

L4: ALTIUM

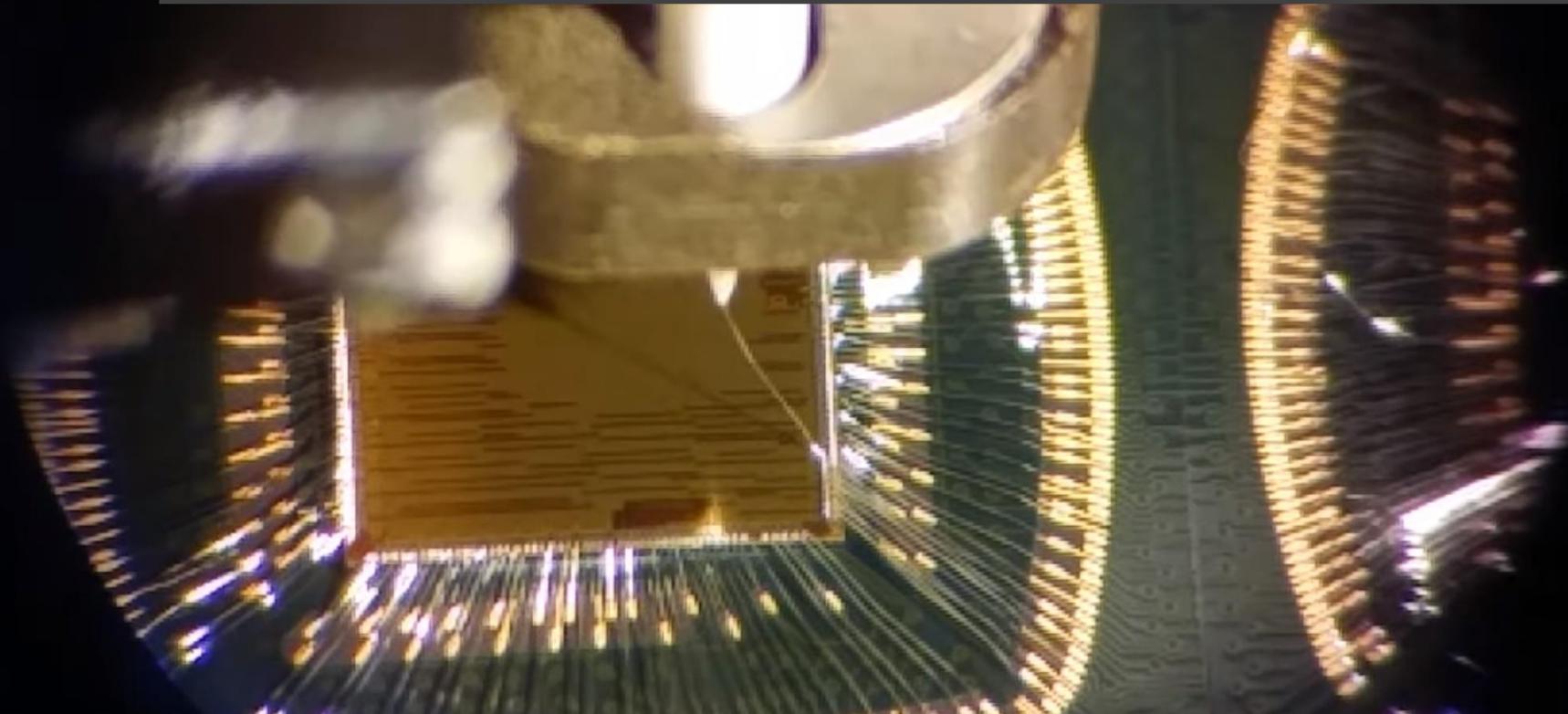
ESE516: IoT Edge Computing

Monday February 4, 2019

Eduardo Garcia - edgarc@seas.upenn.edu

Press Esc to exit full screen

WIRE BONDING



<https://www.youtube.com/watch?v=52IJrf0zdqA>

<https://www.youtube.com/watch?v=3hjCJqsNEIg>

TODAY'S LECTURE

LECTURE GOALS



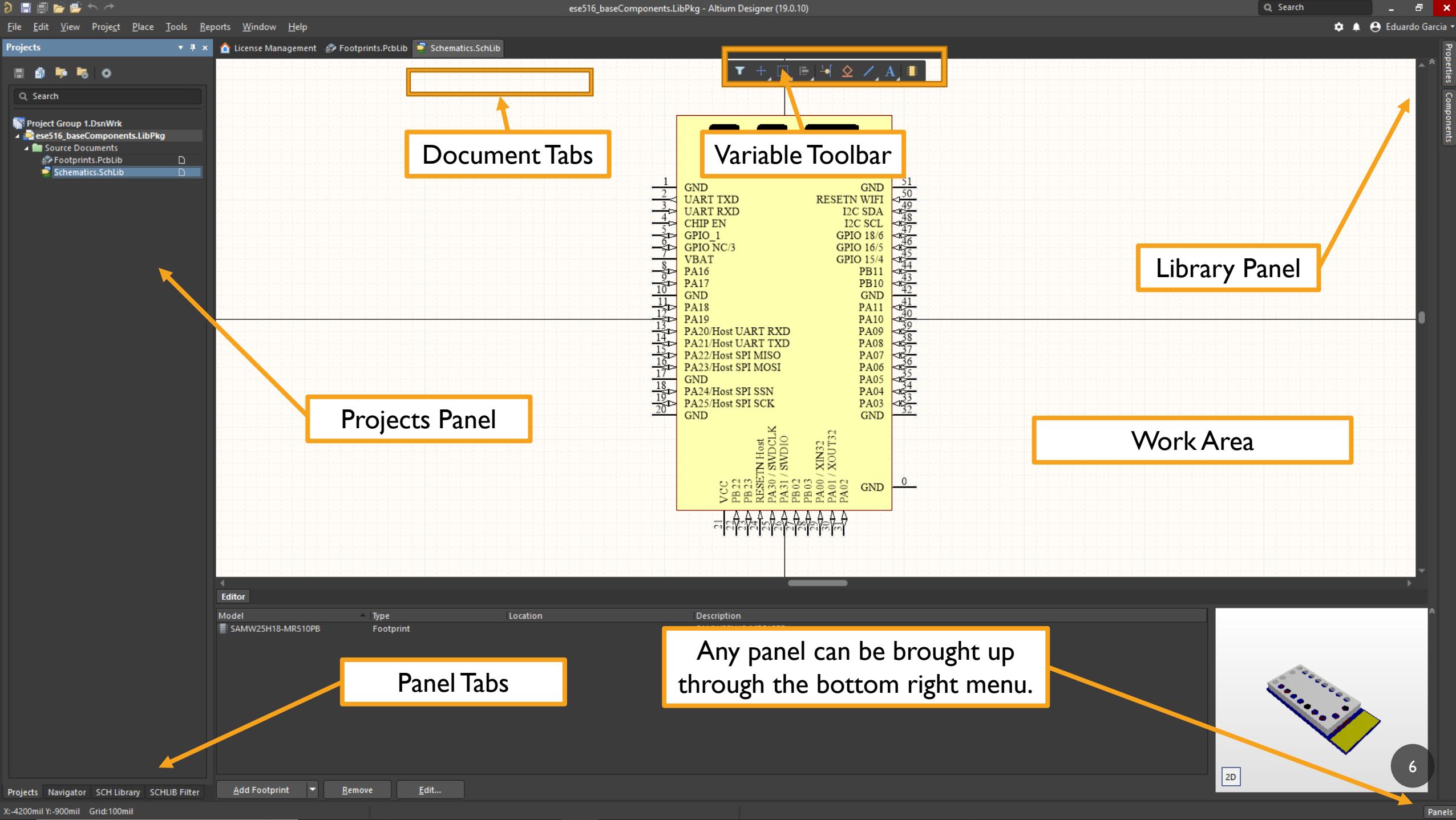
- Introduction to Altium
- Altium Lab: Making Components

ALTIUM

Altium *Designer*®

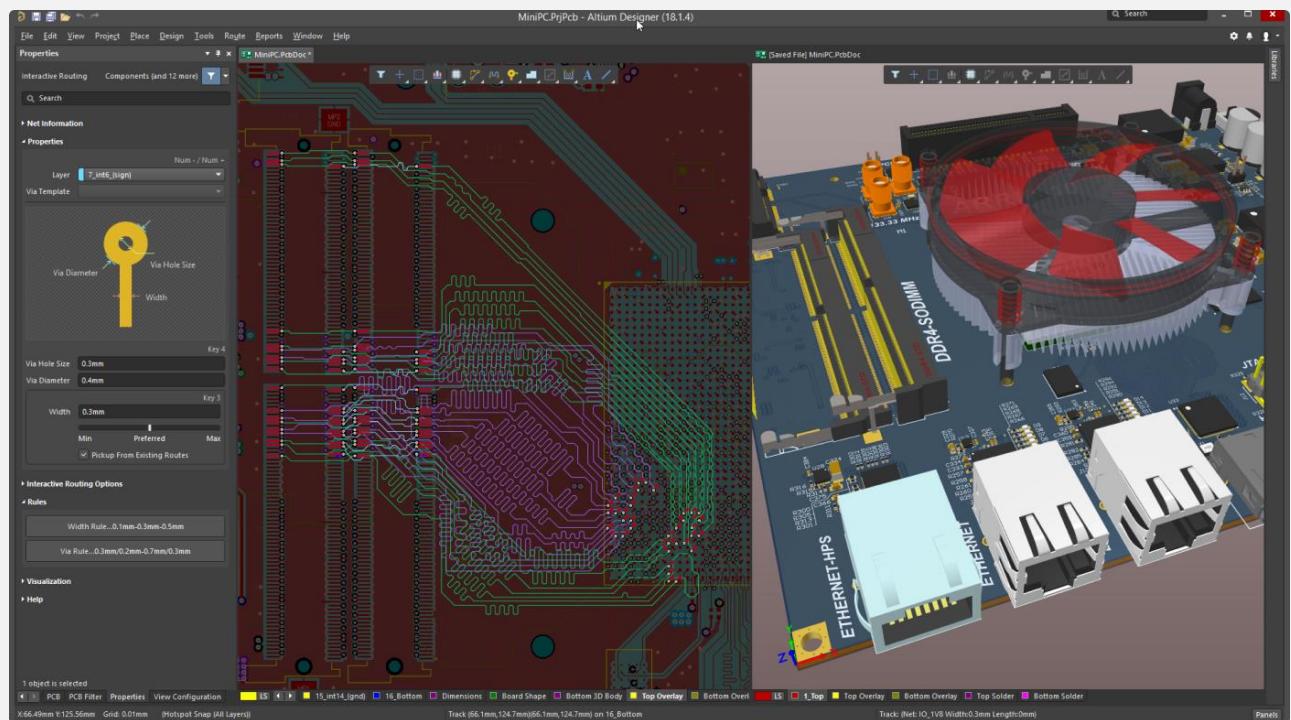


- Altium is a software that allows us to design circuits and fabricate Printed Circuit Boards (PCBs).
- Although we will be learning to use Altium, most of the ideas behind what we will learn can translate to other board design software.
- We will also focus on DFM and best practices of PCBs, which are SW-Agnostic.



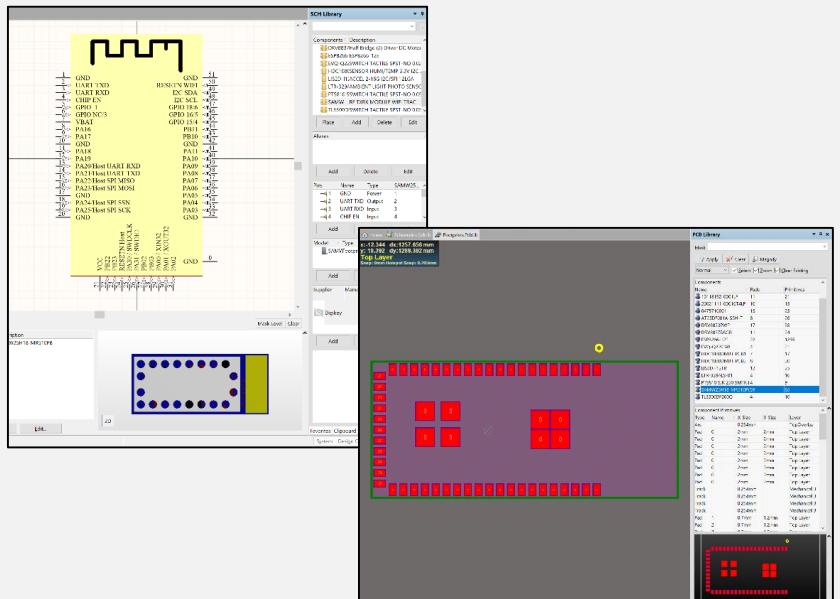
GENERAL FLOW OF PCB DESIGN

- Once we have an idea on what to do, the general flow of a PCB design is the following:
- Component Creation:**
The designers generate the different components to be in the PCB into Altium Designer.
- Schematics:** The designers draw the circuit diagram in schematics files
- Board Layout:** Once the circuit is designed as an schematic, the designers do the physical layout of the board (PCB).

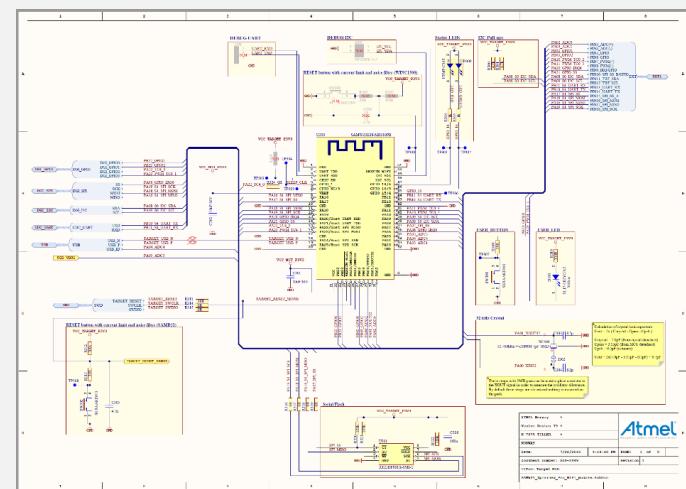


GENERAL FLOW OF PCB DESIGN

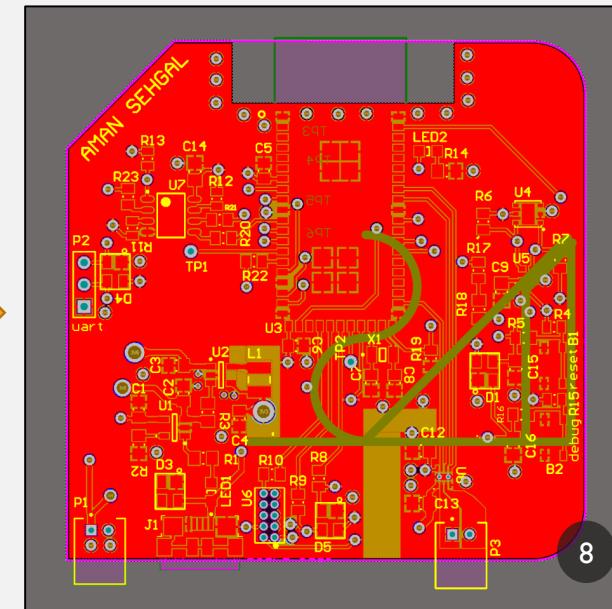
Component Libraries



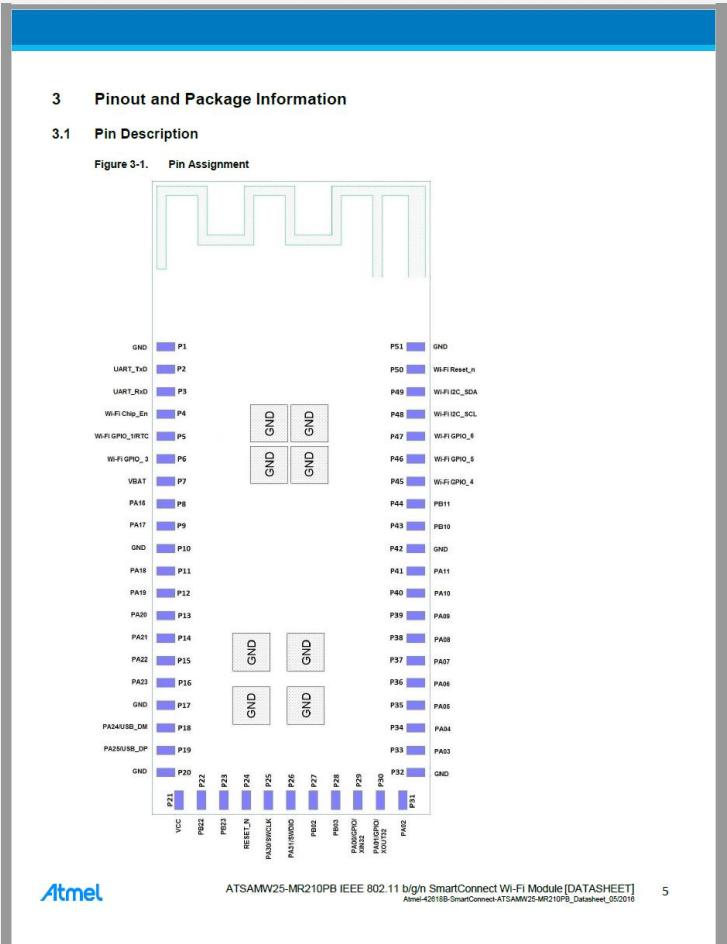
Schematic Sheet



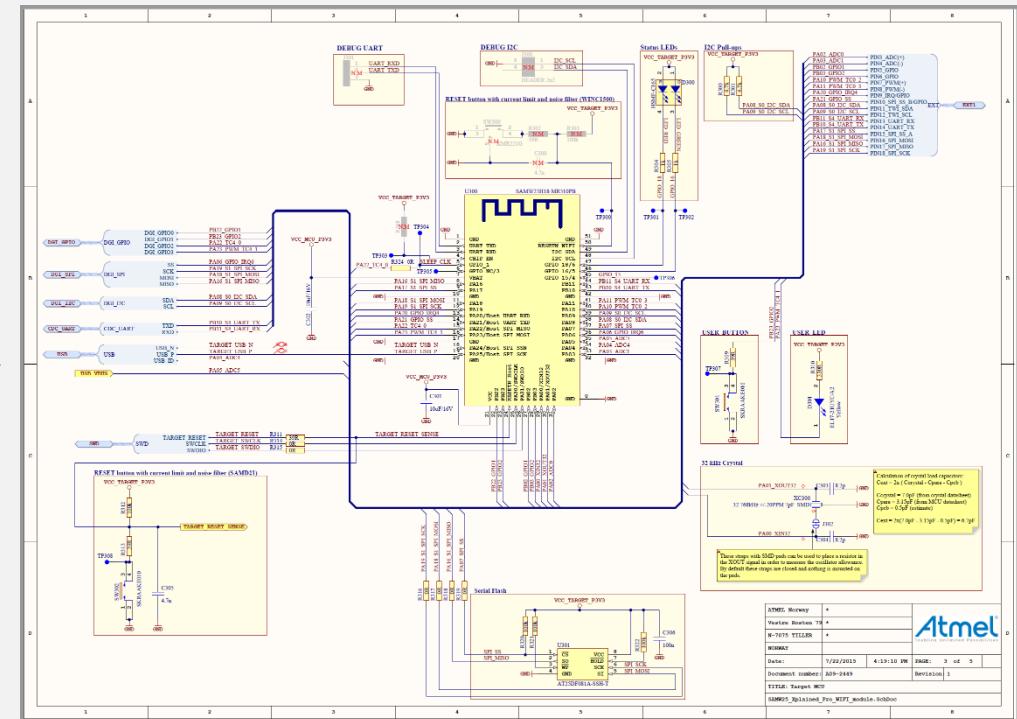
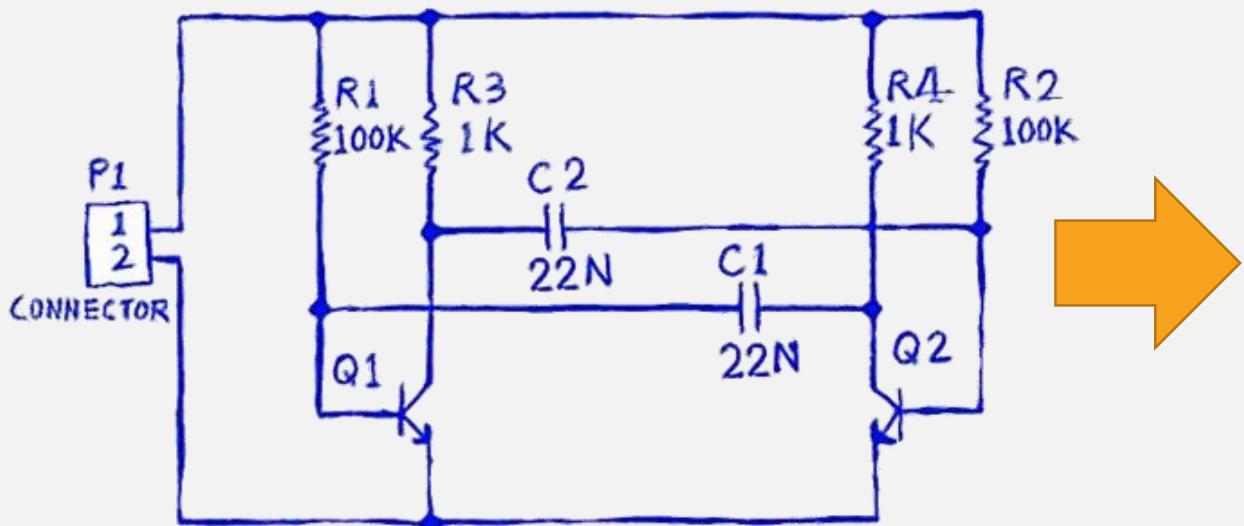
PCB Document



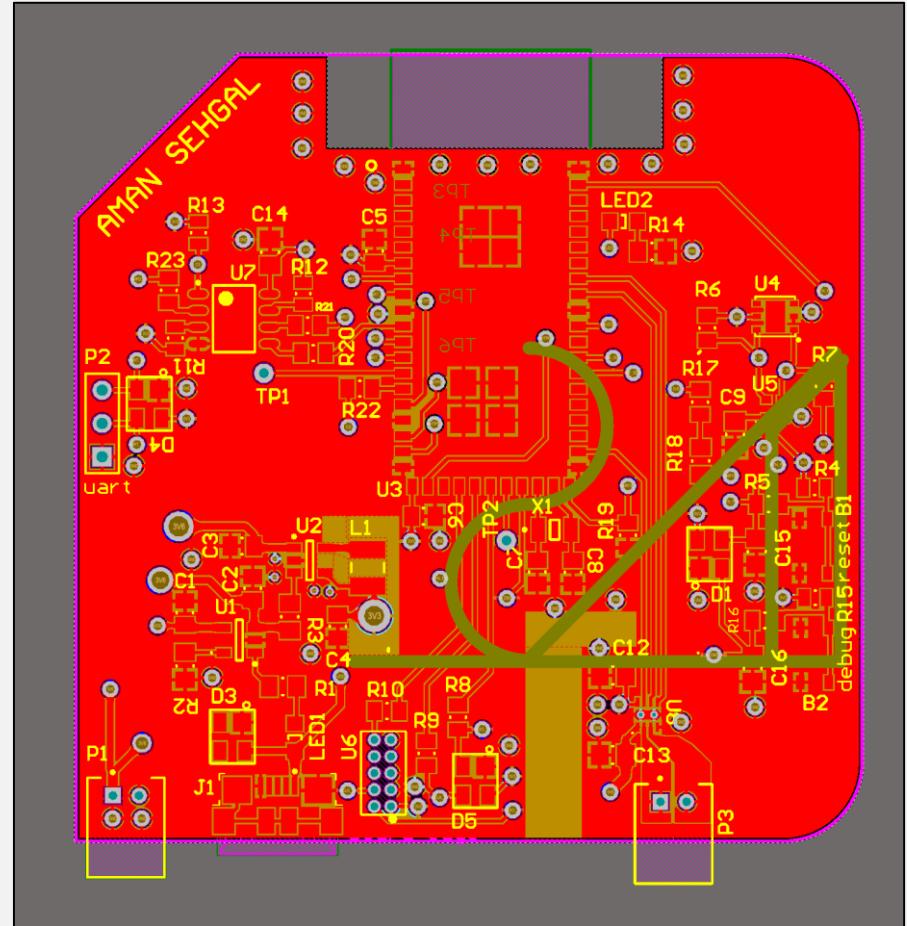
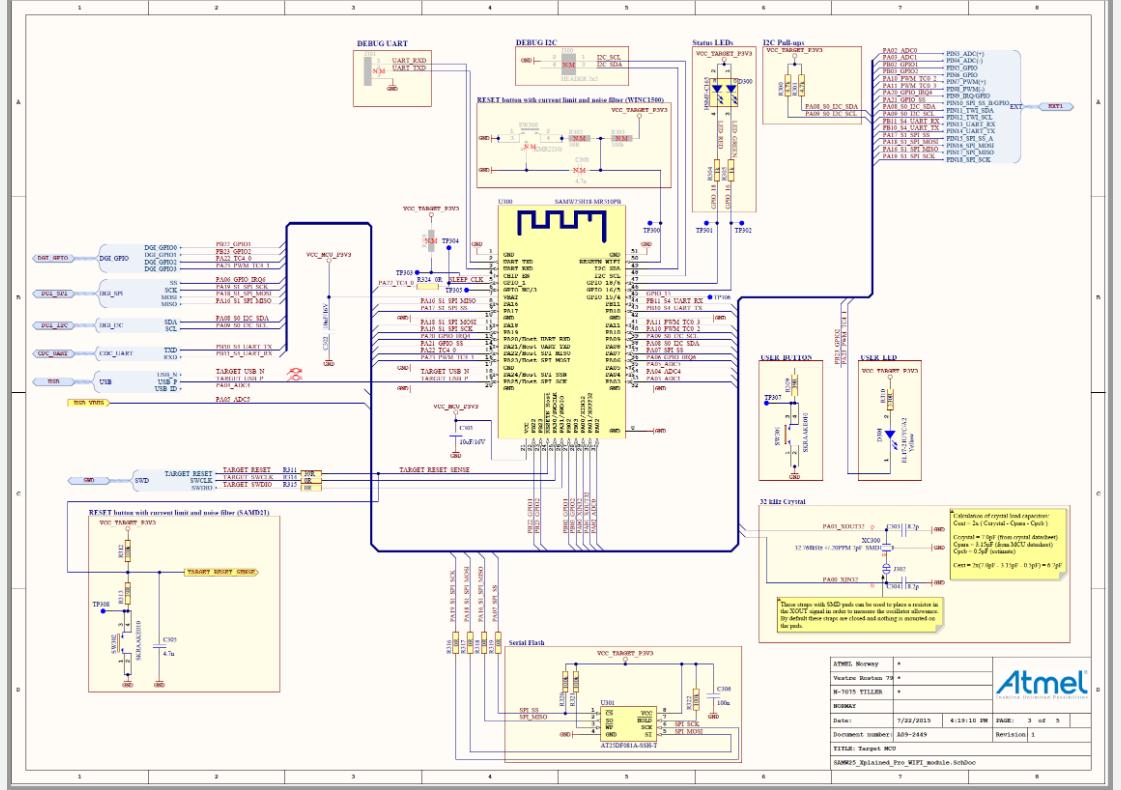
PART CREATION



SCHEMATIC CAPTURE



PCB LAYOUT



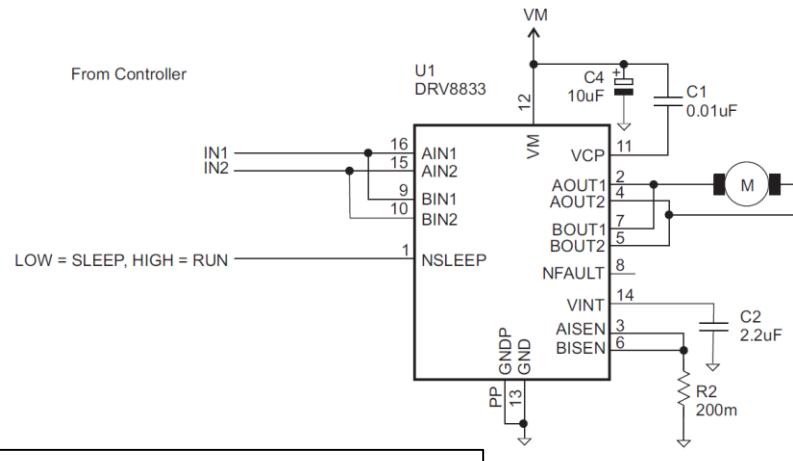
PART CREATION TIPS (BEFORE WE START THE LAB)

TIP #1: DATASHEETS + APP NOTES

- Please refer to them! Most of your questions will be answered already.
- Datasheets and App notes usually contain:
 - Footprint and Pin Assignment
 - Typical application circuits
 - Recommendations for good operation
 - PCB Layout examples

8.2 Typical Application

The two H-bridges in the DRV8833 can be connected in parallel for double the current of a single H-bridge. The internal dead time in the DRV8833 prevents any risk of cross-conduction (shoot-through) between the two bridges due to timing differences between the two bridges. [Figure 7](#) shows the connections.



9 Power Supply Recommendations

9.1 Bulk Capacitance

Having an appropriate local bulk capacitance is an important factor in motor drive system design. It is generally beneficial to have more bulk capacitance, while the disadvantages are increased cost and weight.

The amount of local capacitance needed depends on a variety of factors, including:

- The highest current required by the motor system
- The capacitance and ability to source current
- The amount of parasitic inductance between the power supply and motor system
- The acceptable voltage ripple
- The type of motor used (brushed DC, brushless DC, stepper)
- The motor braking method

The inductance between the power supply and the motor drive system limits the power supply. If the local bulk capacitance is too small, the system responds slowly or dumps power from the motor with a change in voltage. When adequate bulk capacitance is present, the system remains stable and high current can be quickly supplied.

The data sheet generally provides a recommended value, but system-level testing is required to determine the appropriate sized bulk capacitor.

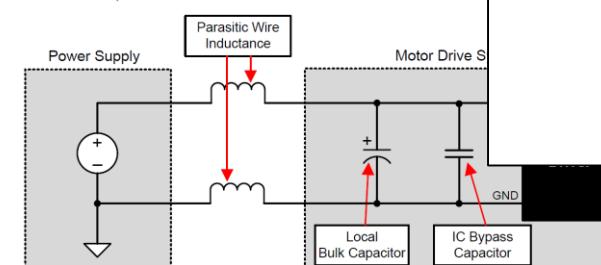


Figure 9. Example Setup of Motor Drive System With External Power Supply

The voltage rating for bulk capacitors should be higher than the operating voltage, to provide margin for cases when the motor transfers energy to the supply.

Iode

10.2 Layout Example

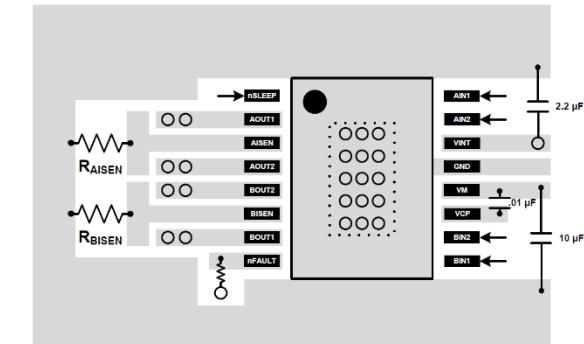
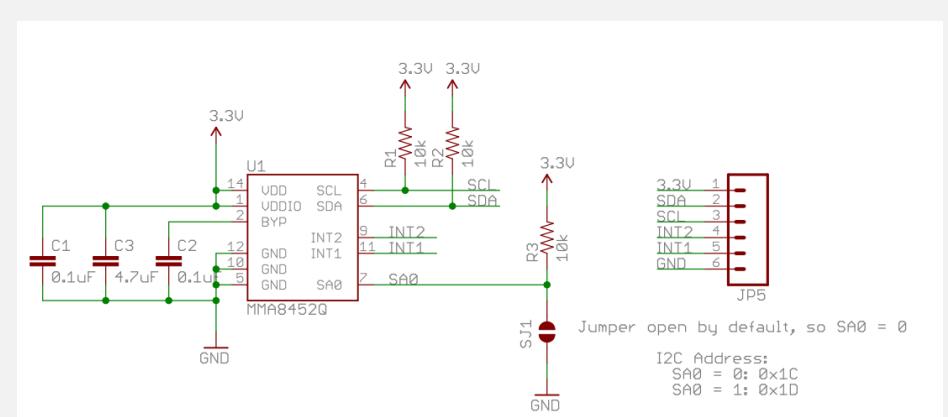
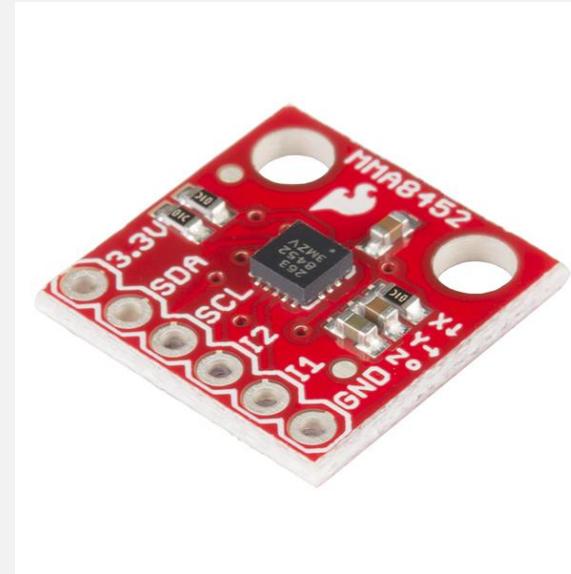


Figure 10. Recommended Layout Example

TIP #2: LEVERAGE DEV KITS!

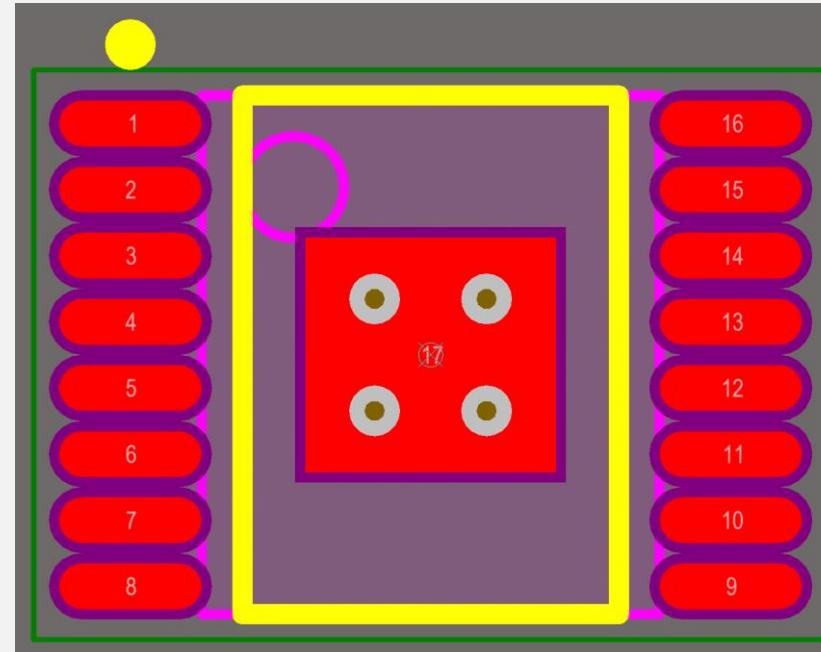
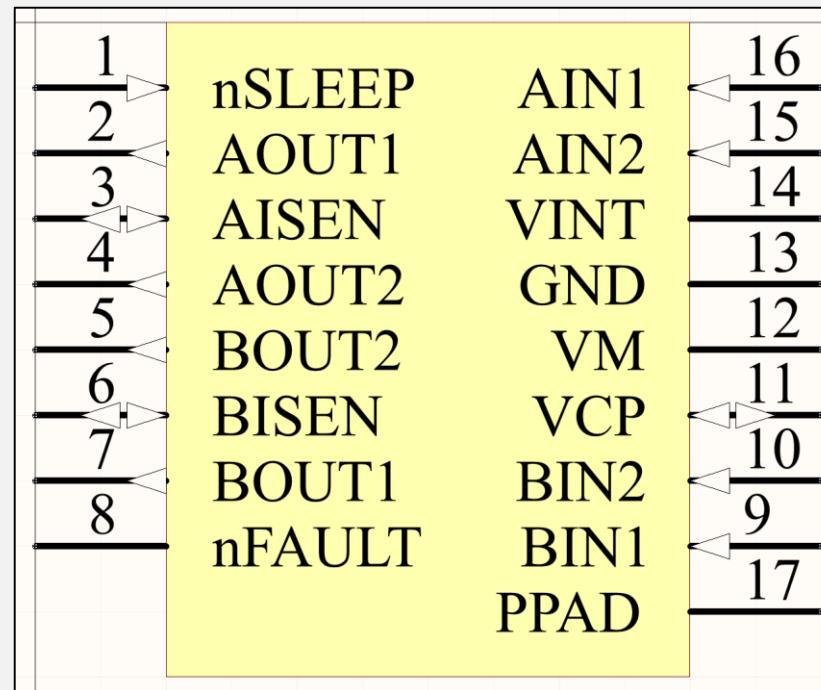
- Dev kits done by the manufacturer or third parties (Adafruit, Sparkfun) have already gone through the effort of prototyping and are selling a product that works.
- Usually the vendor provides the schematics of their dev kit, so you can open these and see how they made their circuit with the sensor you need.
- They may even have Altium projects or footprints for components, but that is rare!



PART CREATION ON ALTIUM

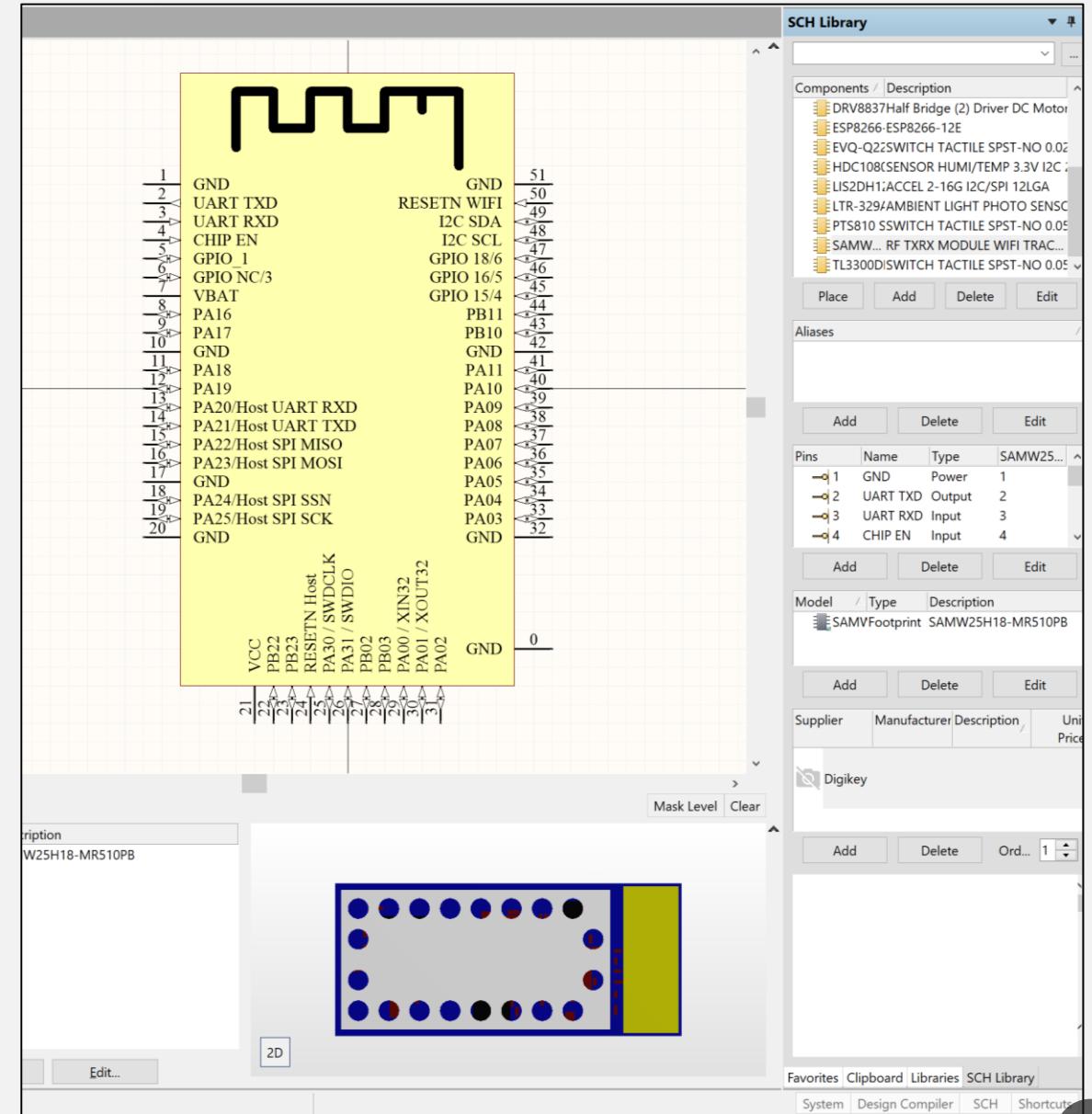
COMPONENTS

- At its most basic, consists of:
 - Schematic symbol
 - PCB Footprint
- Can link other metadata
 - 3D model
 - Cost & stock information
 - SPICE & IBIS models
 - Signal integrity
- Can have multiple PCB footprints
 - Some components come in different footprints



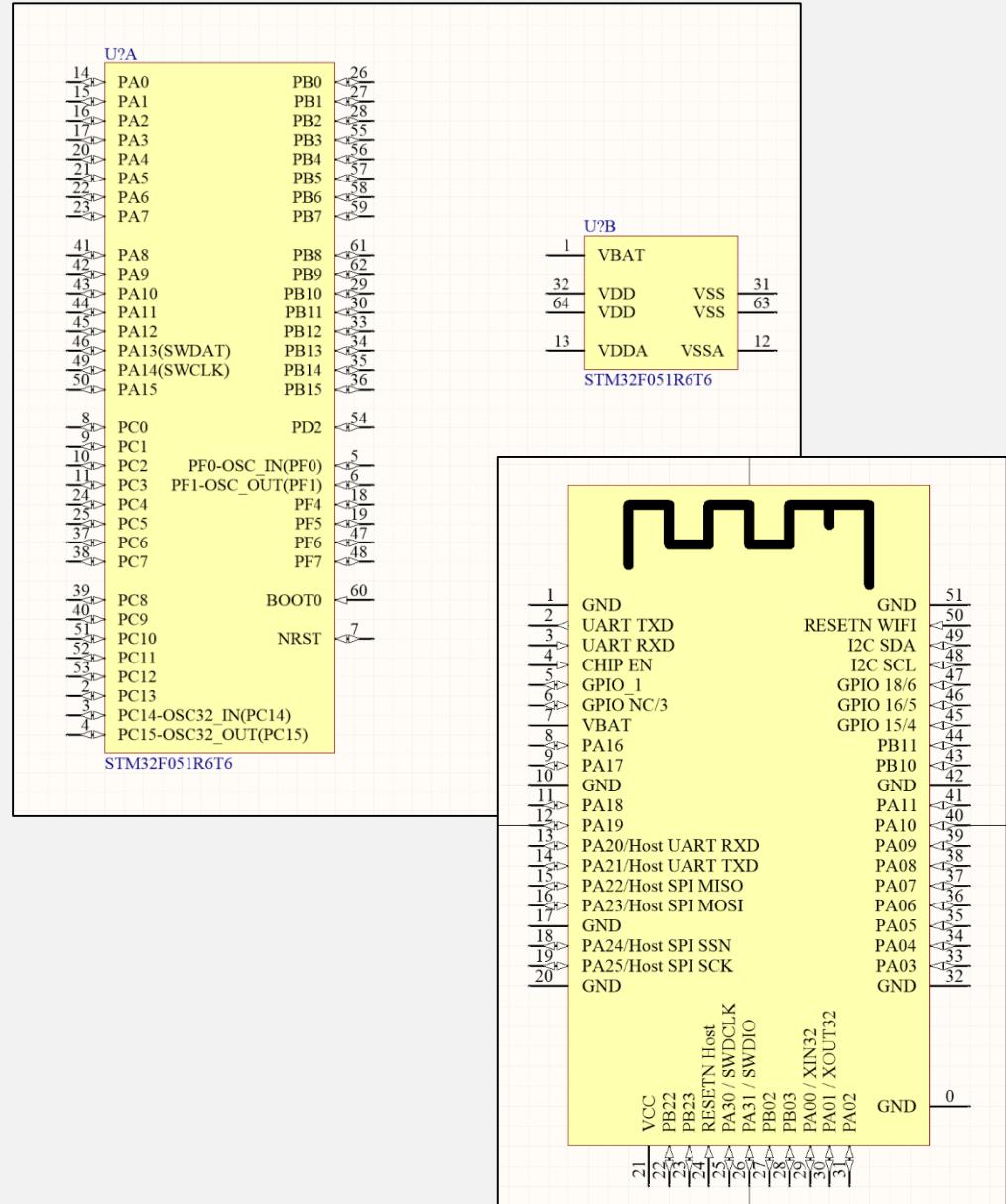
SCHEMATIC LIBRARIES

- A collection of schematic symbols
- Links all models & metadata to the schematic symbol – PCB footprints, SPICE, IBIS, supplier links, etc.



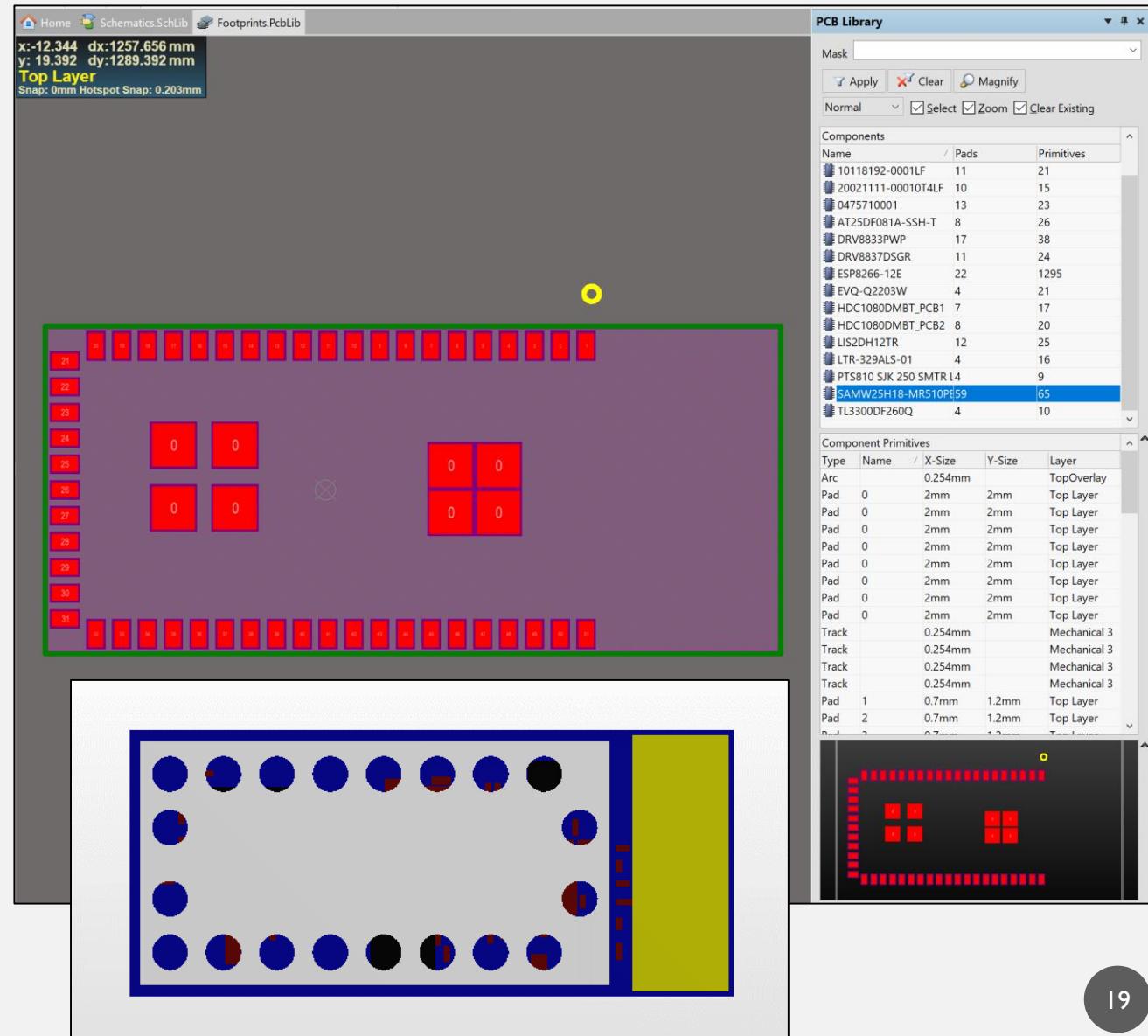
MULTI-PART SCHEMATIC SYMBOLS

- Common to see on MCUs – power and data are separated
- SAM W25 symbol has a physical relation to the actual component – down to the trace antenna
- Either is great! Just depends on preference and readability.



PCB FOOTPRINT LIBRARIES

- A collection of PCB footprints symbols
- 2D information – pad locations, holes, silkscreen, keepouts
- 3D model – great for testing clearances, mechanical engineers love it



HOW TO STORE COMPONENTS?

- We call this a **Local Library** setup.
- For your Altium project, you'll add a **single** Schematic Library and **single** PCB Footprint library.
 - Ensure that you save the library files in the same folder as your other source – makes for a tidier setup.
 - Add and link all of your components between the PCB & Schematic libraries. Make sure to save both of your libraries!
 - Now, you can access your custom library from the **Libraries** panel.
- We recommend NOT using Integrated Libraries! These are used for easy transport of libraries, but you'll only be using these components with your project. You can run into some hiccups with Integrated Libraries.

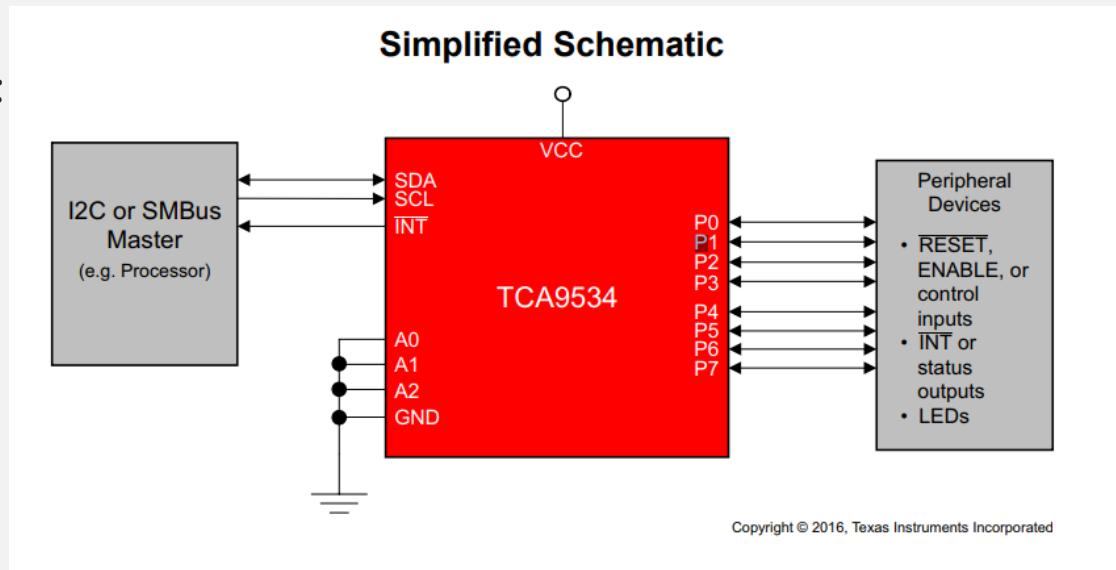
COMPONENT CREATION TUTORIAL

SCHEMATIC LIBRARY

STARTING OUT

- We will make the Altium Part for “TCA9534”. This is a Port Expander that uses I2C to communicate to our MCU. It is useful when we run out of GPIO for our application and we need more GPIO.

- Example use case:



STARTING OUT

- Please open the datasheet for the TCA9534PWR and keep it handy – we will reference this during the tutorial.
- <https://www.digikey.com/product-detail/en/texas-instruments/TCA9534PWR/296-40574-2-ND/5004966>
- Datasheet: <http://www.ti.com/lit/ds/symlink/tca9534.pdf>



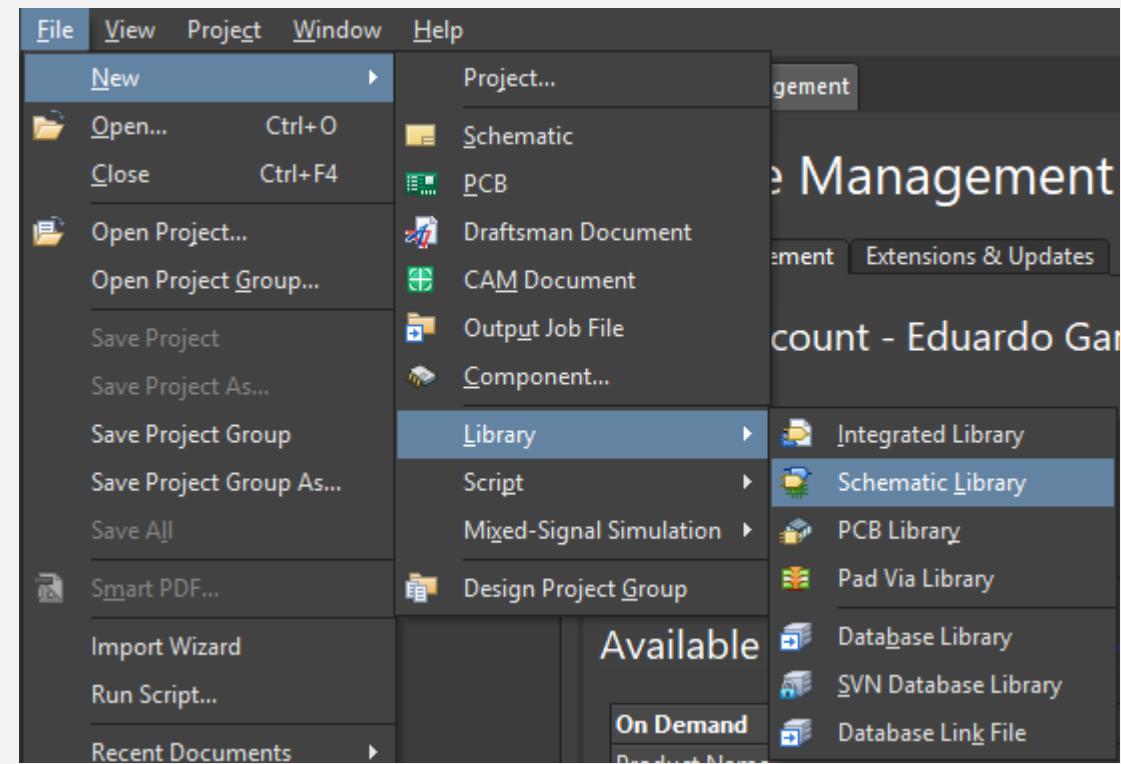
TCA9534

SCPS197D – SEPTEMBER 2014 – REVISED OCTOBER 2017

TCA9534 Low Voltage 8-Bit I²C and SMBUS Low-Power I/O Expander with Interrupt Output and Configuration Registers

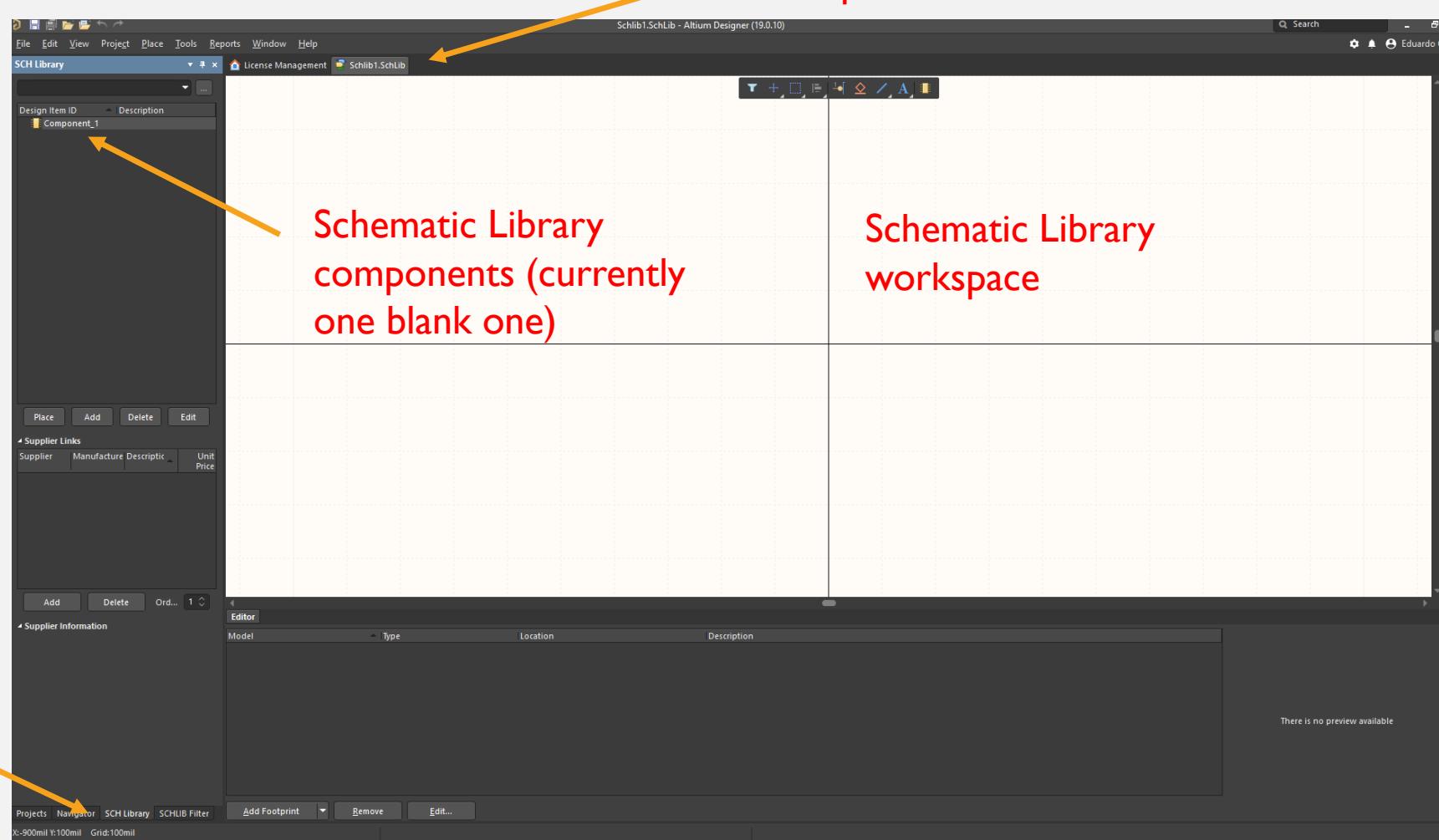
MAKING A SCHEMATIC LIBRARY

- Let's start with a schematic library. The schematic library will hold the symbols for the devices we make for our project.
- Go to “File->New->Library->Schematic Library”



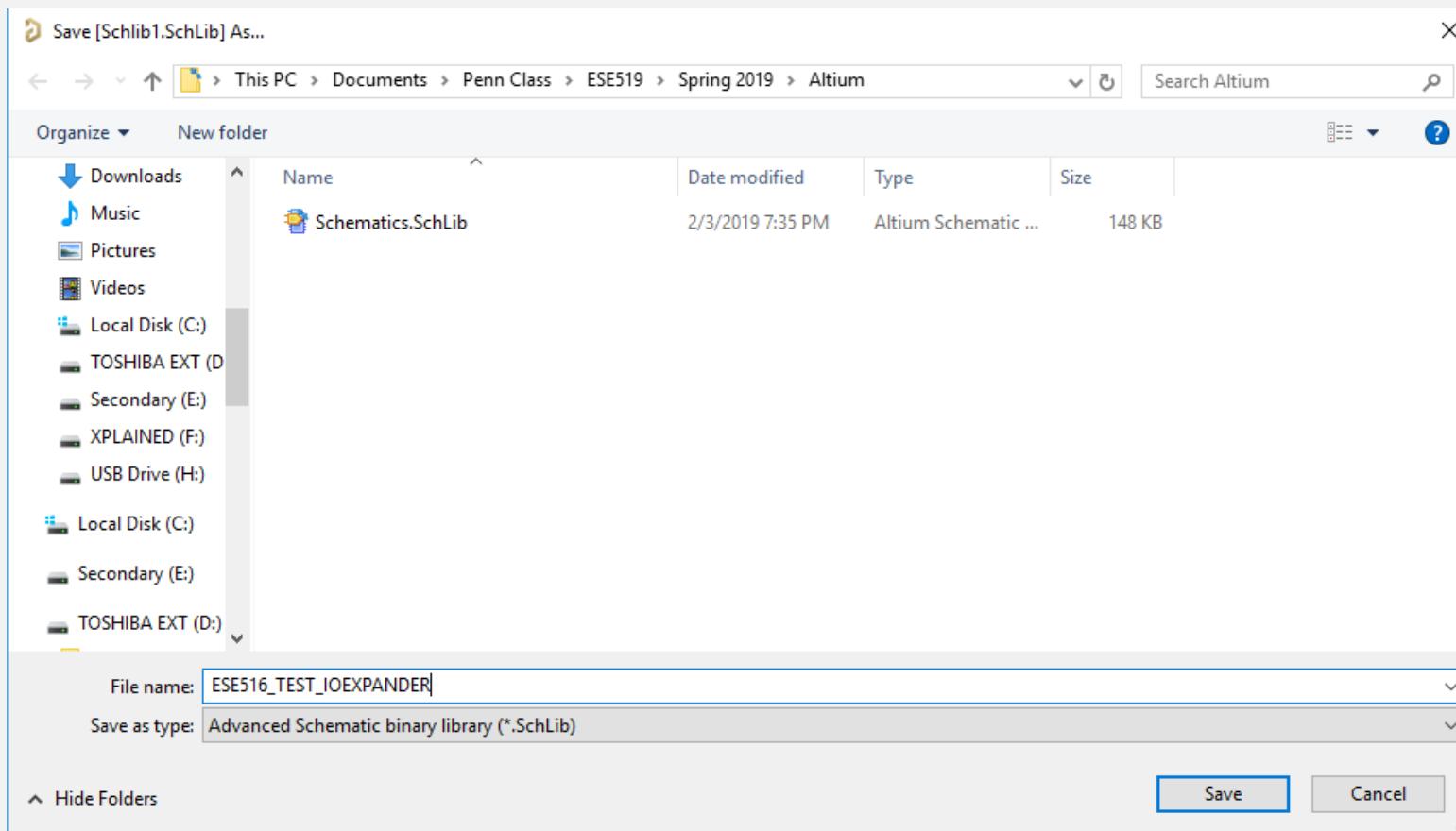
MAKING A SCHEMATIC LIBRARY

- This will open a blank schematic Library.



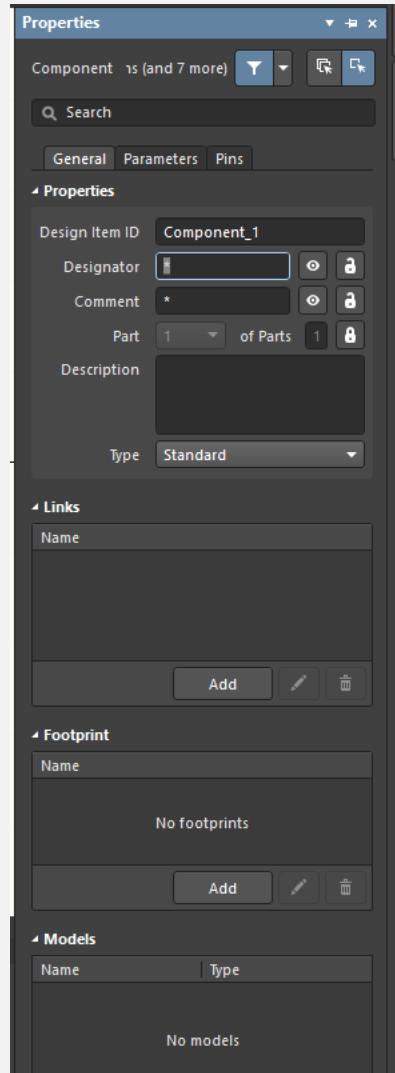
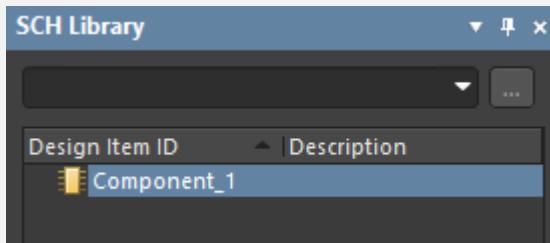
MAKING A SCHEMATIC LIBRARY

- Before we continue, please save the file (File->Save...) and give it any name you would like.



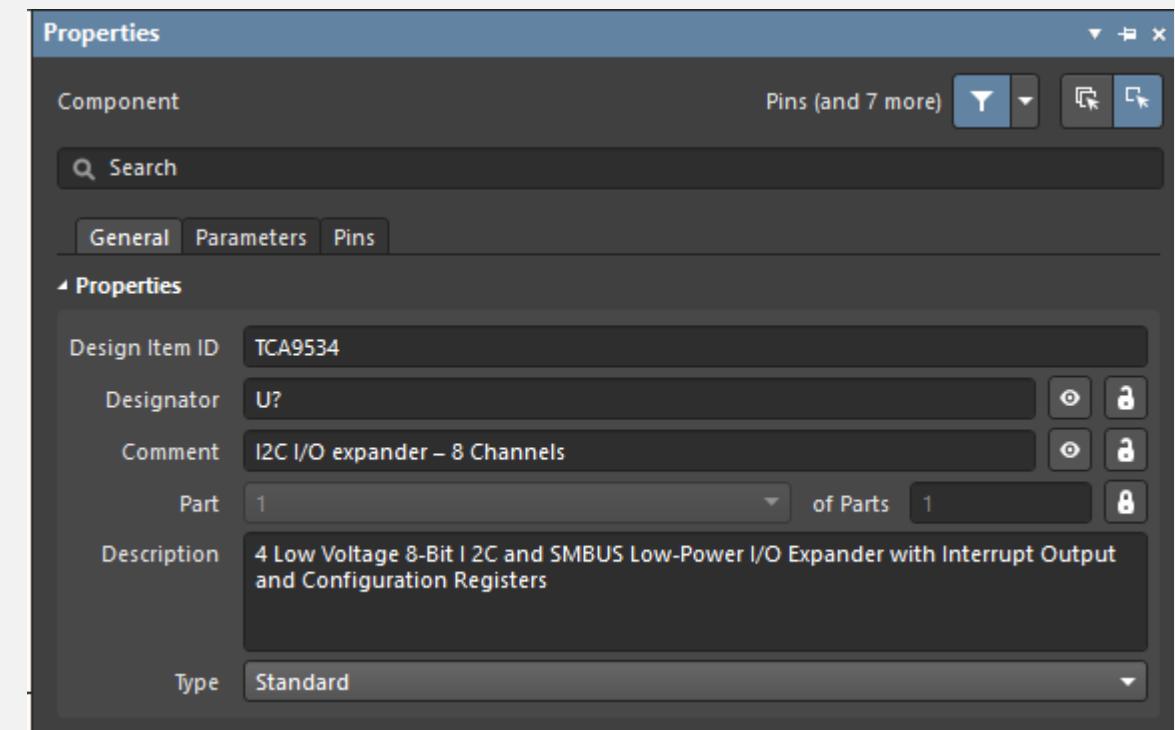
ADDING A COMPONENT TO A SCHEMATIC LIBRARY

- We can add how many components we want to this Schematic Library. By default, Altium adds a blank component, called “Component_1”. We will use this to make our first component.
- Double-click on “Component_1”. A “Properties” tab will appear on the right side of the screen



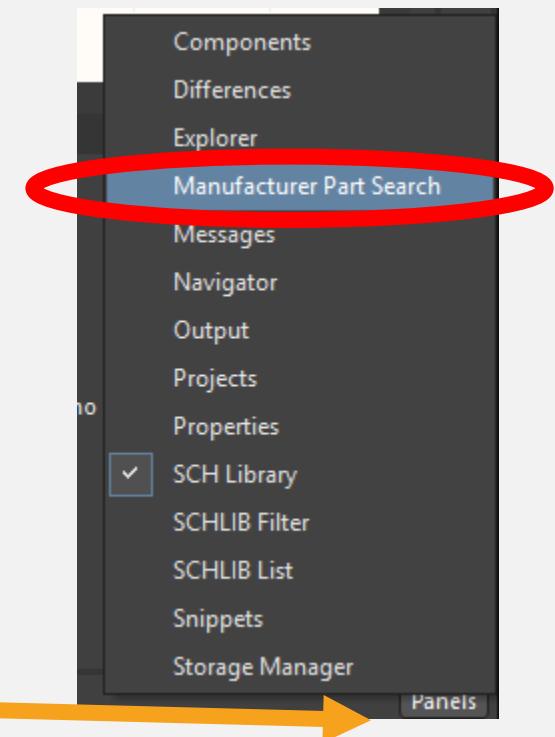
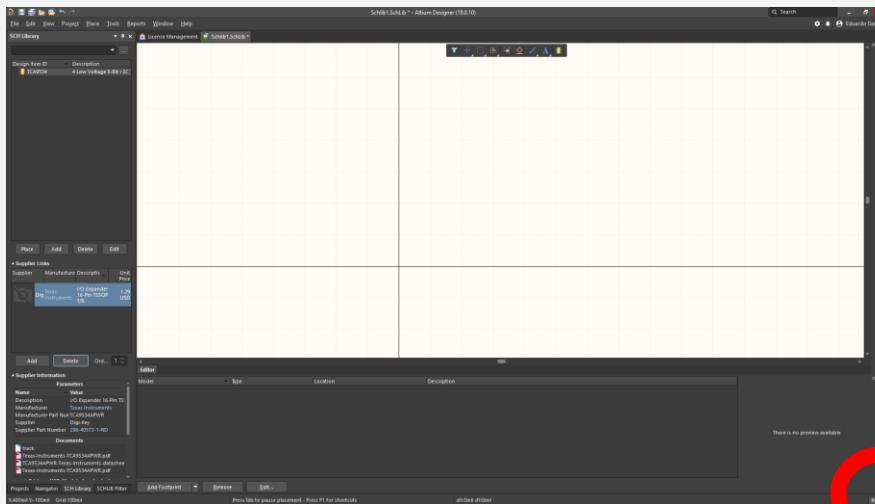
ADDING A COMPONENT TO A SCHEMATIC LIBRARY

- We now need to edit the properties of this blank component to reflect the component we want to create. **It is good practice to fill out as much information as possible on the component for future re-use!**
 - On “Design Item ID” we can add the name of the component - TCA9534
 - This is an IC, which usually uses a designator of “U”. This is the style of designator that will be used on the silkscreen on a PCB. Please add “U?” here. This means, the designator will be “U”, followed by a number depending on a unique number dependent on a project.
 - On “Comments” we can add a string that will appear beside the component – let’s add “I2C I/O expander – 8 Channels”
 - On “Description”, it is a good idea to add a description of the component. Please add this string: “Low Voltage 8-Bit I 2C and SMBUS Low-Power I/O Expander with Interrupt Output and Configuration Registers”.
 - Leave type as standard – this designates the component type. Some, for example, can be mechanical components.



ADDING METADATA TO A COMPONENT

- We can add other information from the manufacturer/Digikey with the help of Altium! **This is really good practice, please remember to always do this!** It fills important information directly from Digikey, and saves you time from doing it manually.
- Go to the “Panels” Button on the bottom right of the screen, and choose “Manufacturer Part Search”. This will bring a panel where you can search for a device by putting the Manufacturer Part Number (MPN).

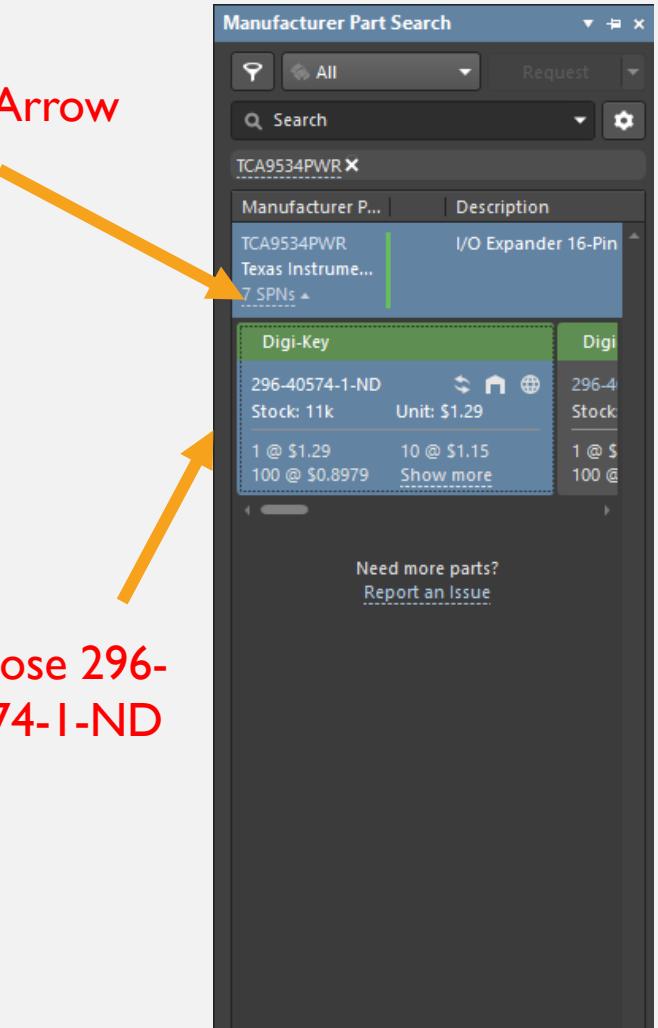


ADDING METADATA TO A COMPONENT

- On the “Manufacturer Part Search”, search for “[TCA9534PWR](#)”.

- The results will appear minimized. Click on the small down arrow to show the results from Digikey (Please always use Digikey for this class when doing components!).
- Right click on the option “296-40574-1-ND” from Digikey. That is the SPM – Supplier Part Number. It is usually different from the MPN and varies from supplier to supplier.

Down Arrow

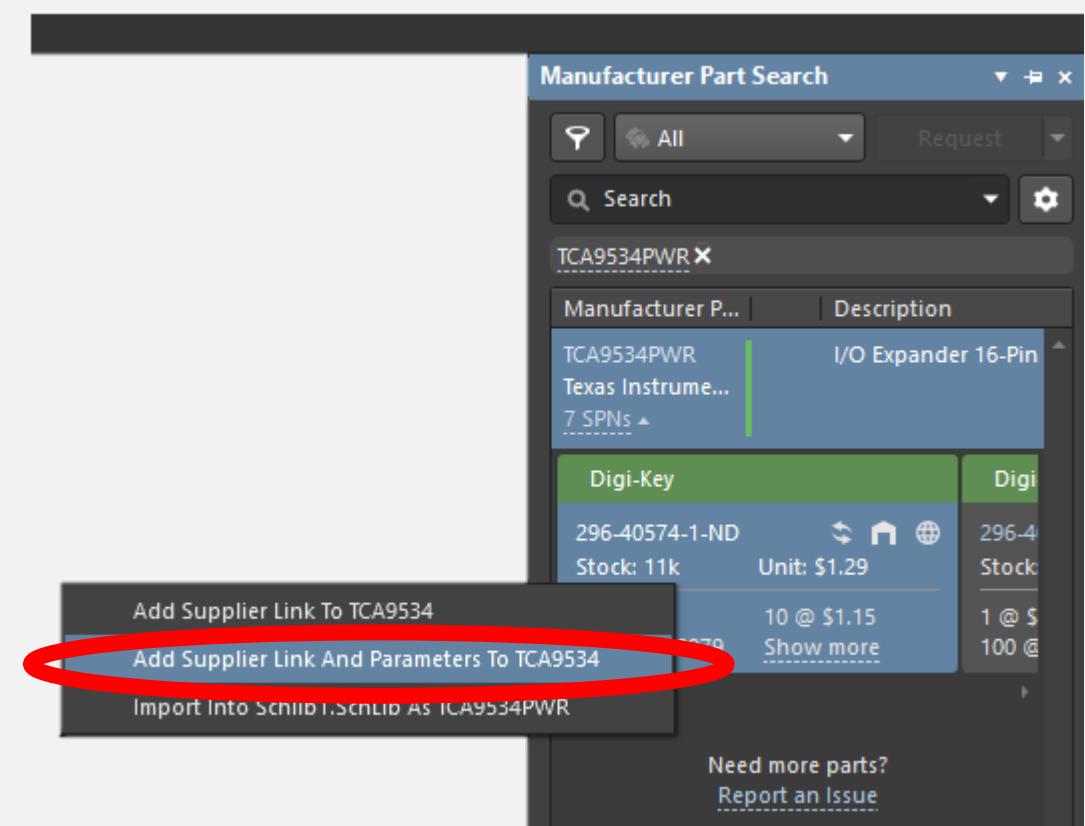
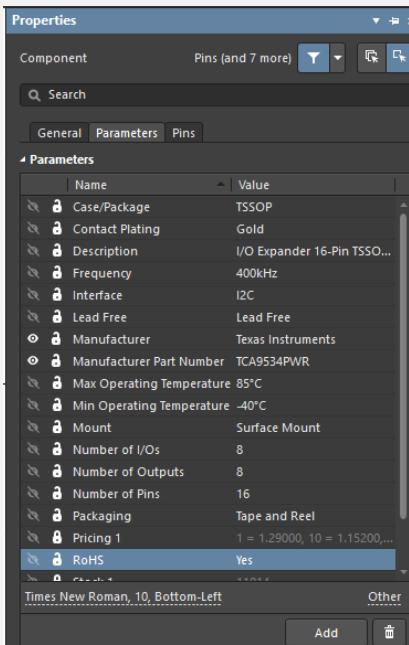


ADDING METADATA TO A COMPONENT

- Right click on this solution, and choose “Add Supplier Link and Parameters to TCA9534”

- This will add a lot of useful parameters to the metadata of the component – check it out by going to the “Parameters” Tab on the Properties Panel

Useful data
added
automatically
from Digikey!



SYMBOL DRAWING CONSIDERATIONS

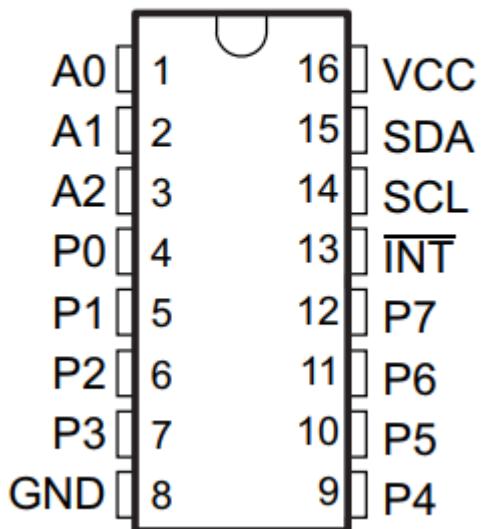
- We are now ready to draw the symbol for a component! Before we begin, we must talk about how to draw a component. There are two schools of thought:
 - I. **Draw is as it looks from above:** This draws a component identically as it is physically. It puts the pins of the component on the same order than the physical component. It makes verification of a PCB with respect to a physical component easier, but it makes drawing components hard!
 2. **Draw the component according to function :** This draws the component in a way that better explains its behavior, rather than how it is really physically. The drawing will differ from the real component, but it will make drawing schematics much clearer!

SYMBOL DRAWING CONSIDERATIONS

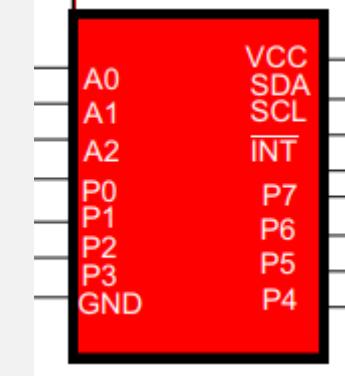
- Example:

- Physical IC

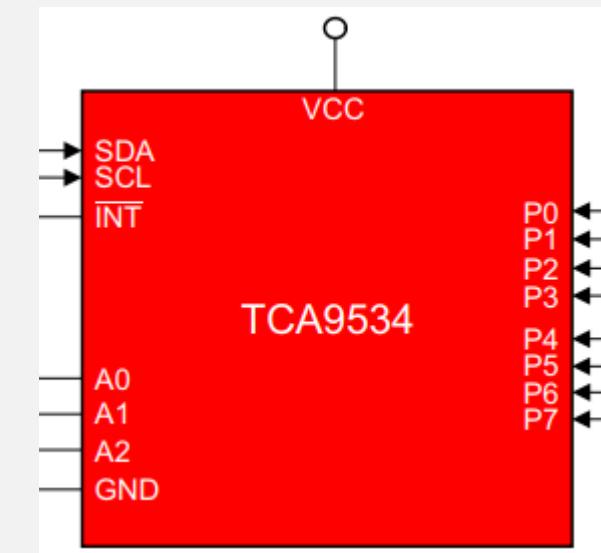
PW, DW Package
16-Pin TSSOP, SOIC
Top View



Drawing schematic 1:1 to physical



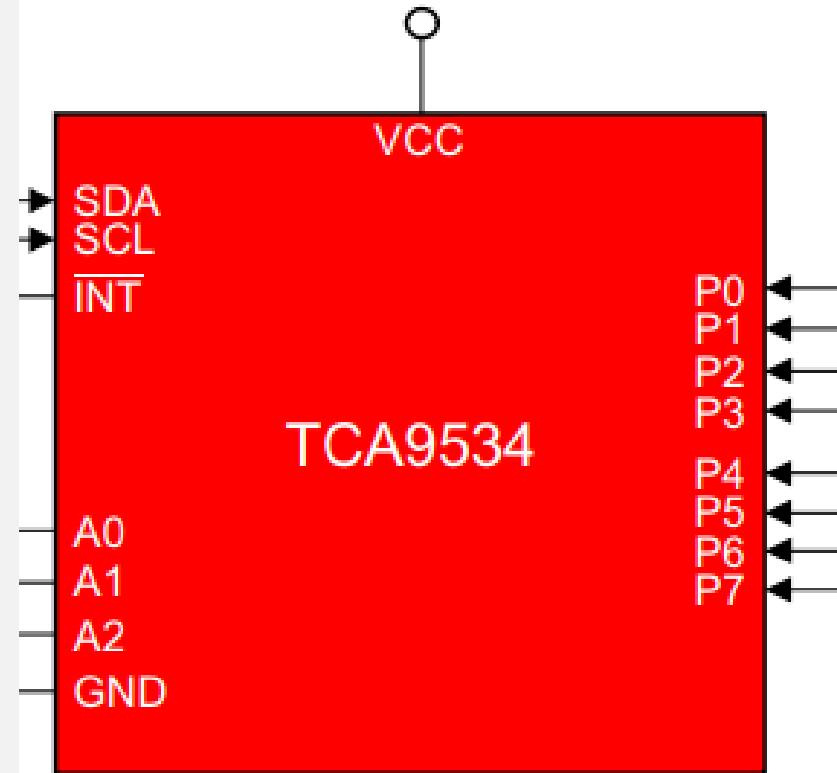
Drawing schematic by function



SYMBOL DRAWING CONSIDERATIONS

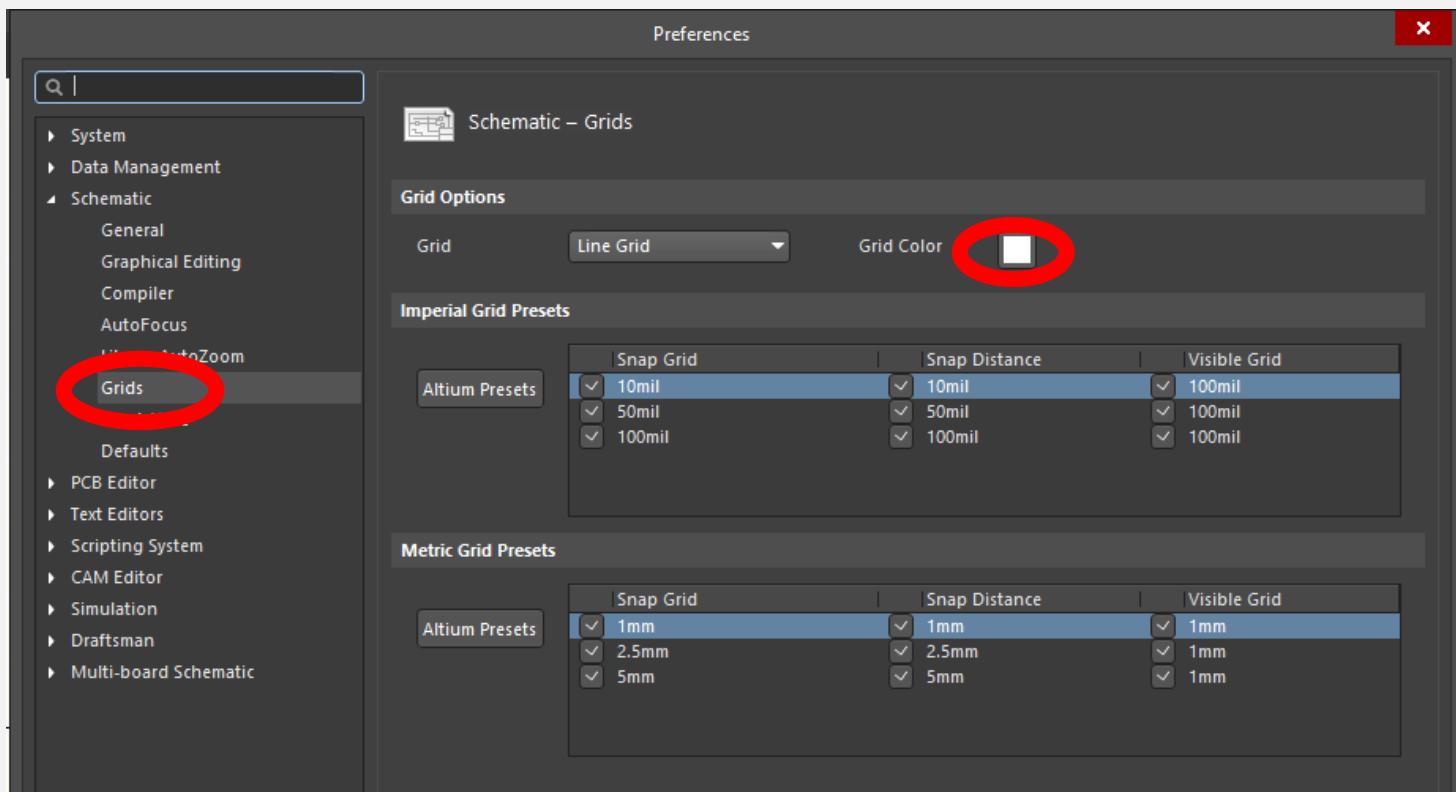
- For this class (and in industry in general) we will prefer to **Draw the component according to function**
- Some tips for doing this:
 - Draw input /Comm channels on the left, Outputs on the right
 - Separate different parts of the circuit, join similar parts
 - Example: Join the I²C channels, Join the I/O Output channels, Separate I²C and I/O Channels
 - **Tip:** The datasheet usually has a very good recommendation on how to draw the symbol – I usually follow their direction or tweak it to my needs. In this example, the TCA9534 has a great symbol and we will re-use it.

Simplified Schematic



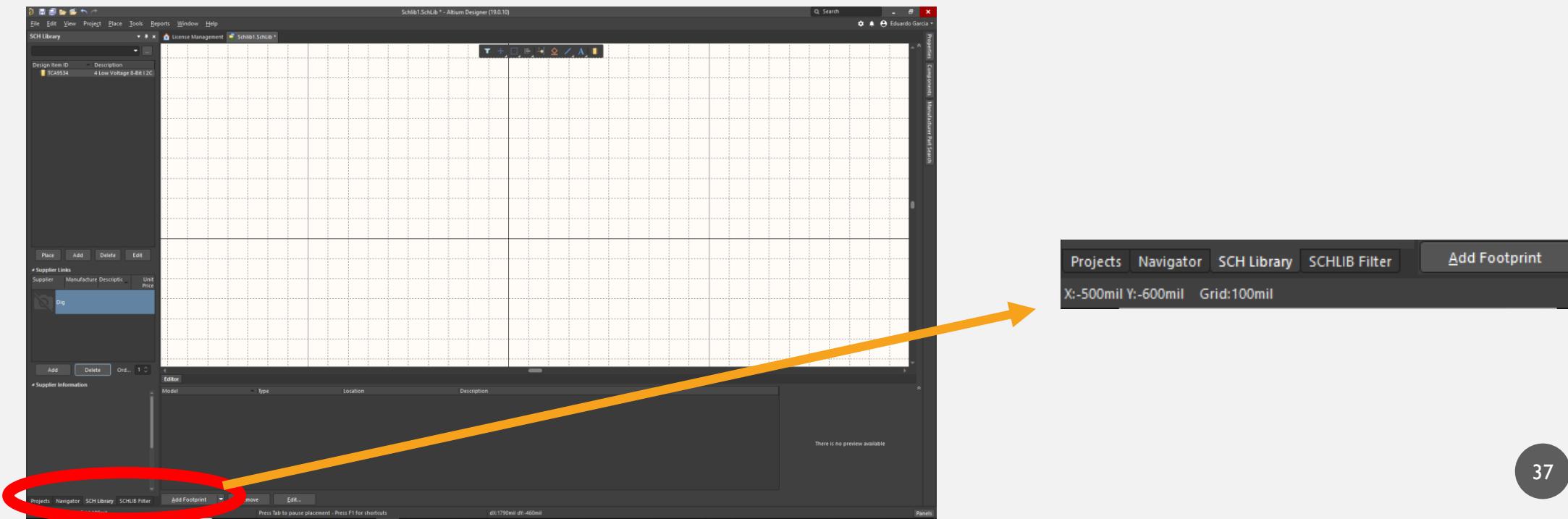
FIXING THE GRID

- Before we draw, let's do some tweaks to our environment. Let's set the grid to something visible, and check the grid snaps.
- Right click on the workspace and choose “Preferences...”.
- Choose “Grids” then select the Grid Color and hit “Ok”. I recommend some grey color.



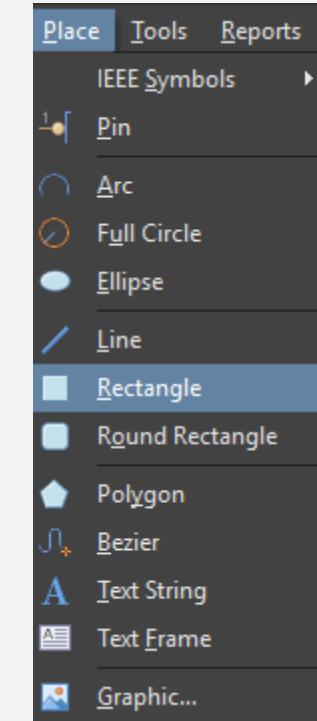
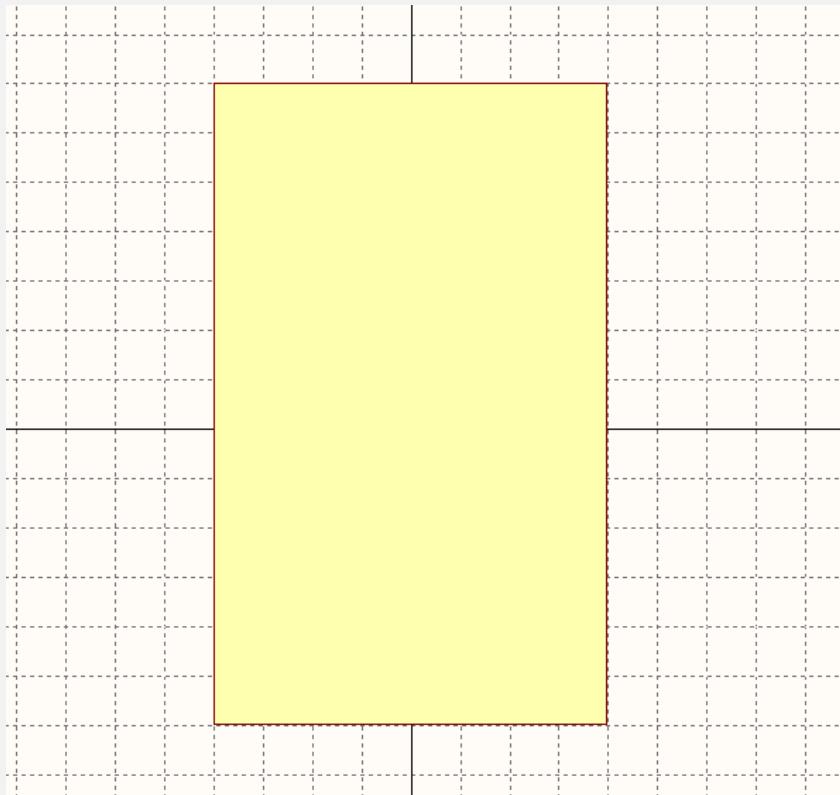
NOTES ABOUT THE GRID

- To see the snap grid you are currently using, check the bottom bar of Altium – It will tell you the X and Y coordinates where the mouse currently is as well as the grid size.
- To toggle through the grid sizes, press “g” (keyboard shortcut).



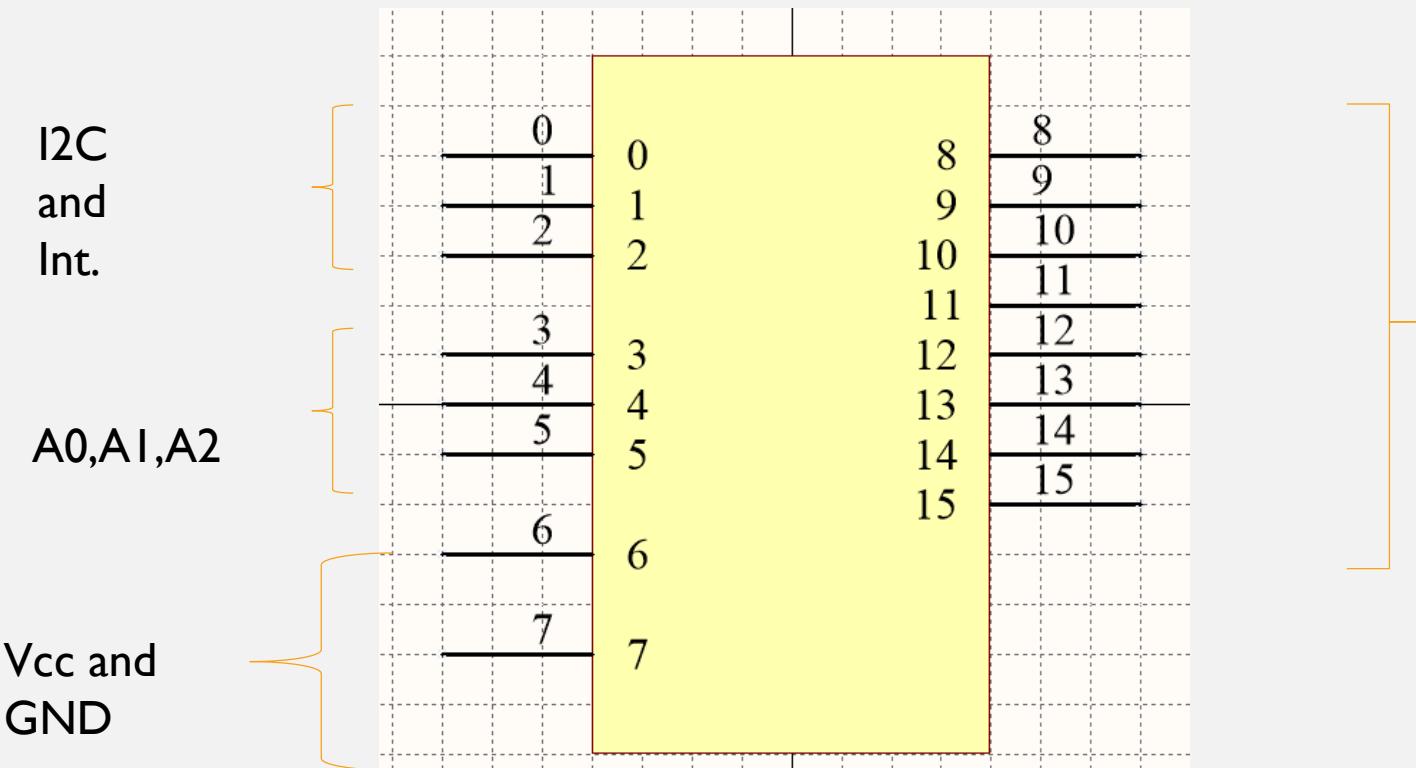
DRAWING THE COMPONENT-MAKING THE BODY

- Go to “Place->Rectangle” and draw a rectangle similar to the one below.
We can change the dimension later

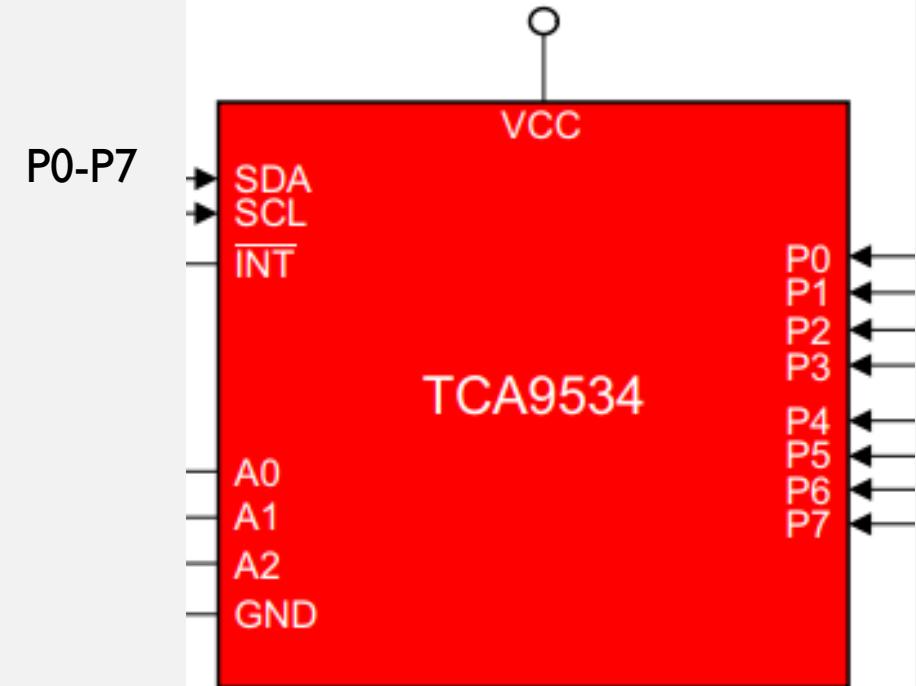


DRAWING THE COMPONENT – ADDING PINS

- We now need to add the pins. To do so, go to “Place-Pin”. This will attach a pin to your cursor. Place 16 pins as shown below. Don’t worry about the specific text on each pin – we will change that later. For now, we are just giving the shape to the device. (Tip: Spacebar rotates Pin).

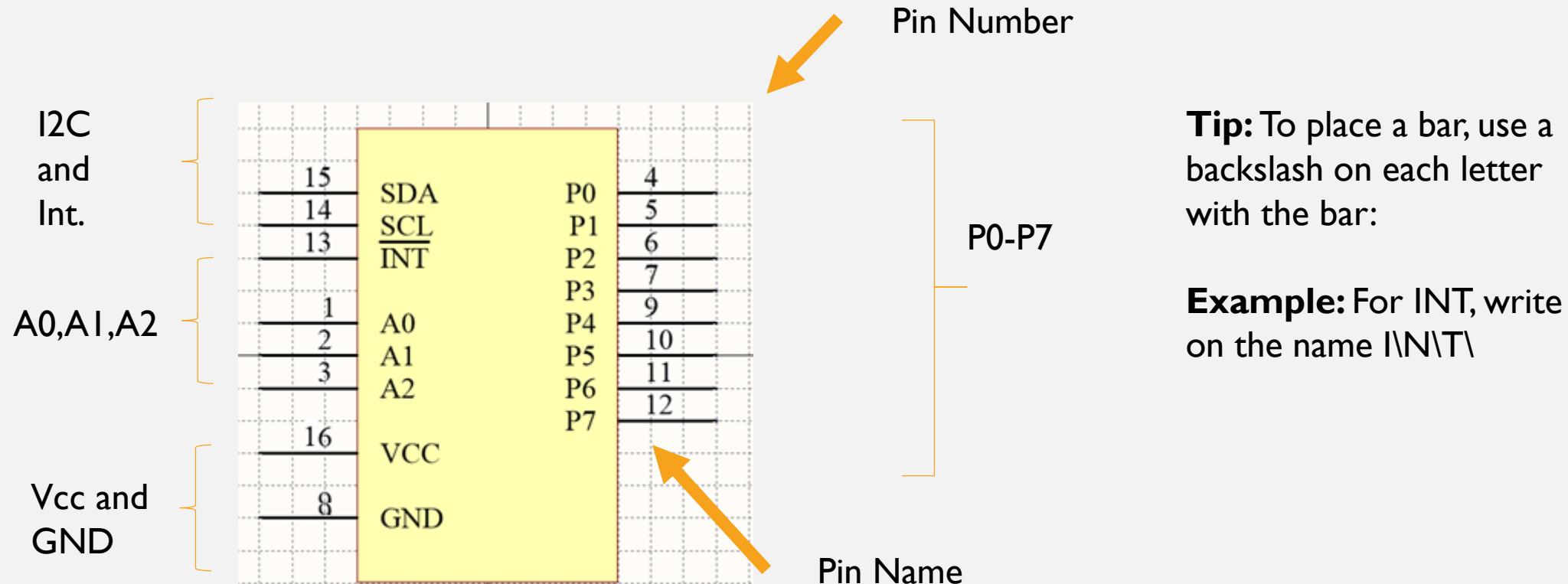


Simplified Schematic



DRAWING THE COMPONENT – ADDING PINS

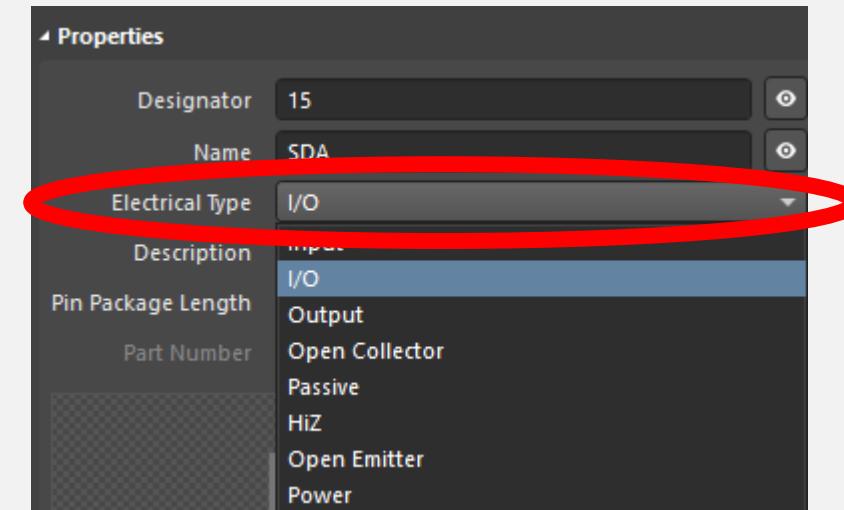
- Now, we need to add the Pin# and the Pin name to each pin. For this, we must use the datasheet and **Double Check** that we are assigning the pin number correctly!
- Double click on each pin and add the Designator (Pin#) and Name (Pin Name). Please add the info to each pin until your device looks like the one below.



DRAWING THE COMPONENT – ADDING PINS

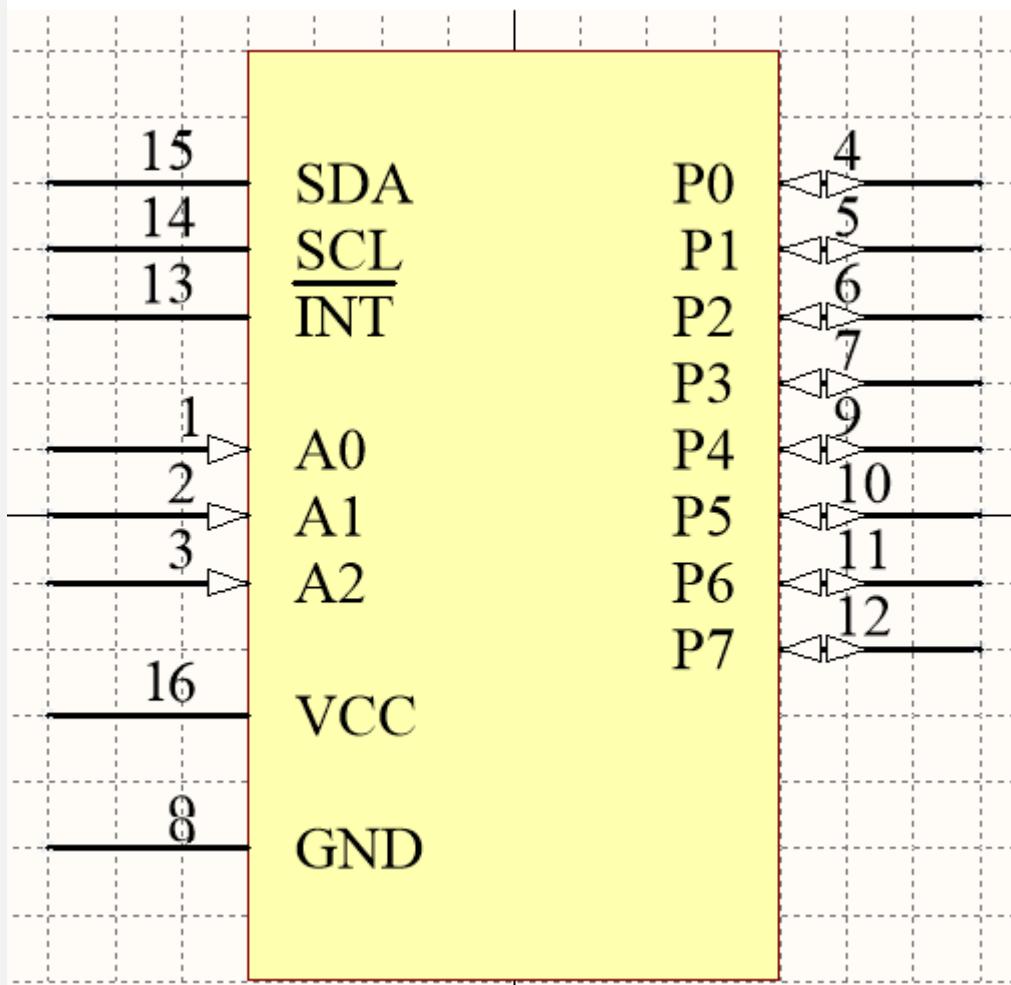
- We can also add to Altium what type of pin do we have – Input, Output, Power, I/O, etc.
- To do this, we double click on the pin and on the Properties panel we choose from the “Electrical Type” dropdown.
- Please select the following for the pins:

- P0-P7: I/O
- A0-A2: Input
- Interrupt, SCL, SDA: Open Collector
- VCC,GND: Power



DRAWING THE COMPONENT – ADDING PINS

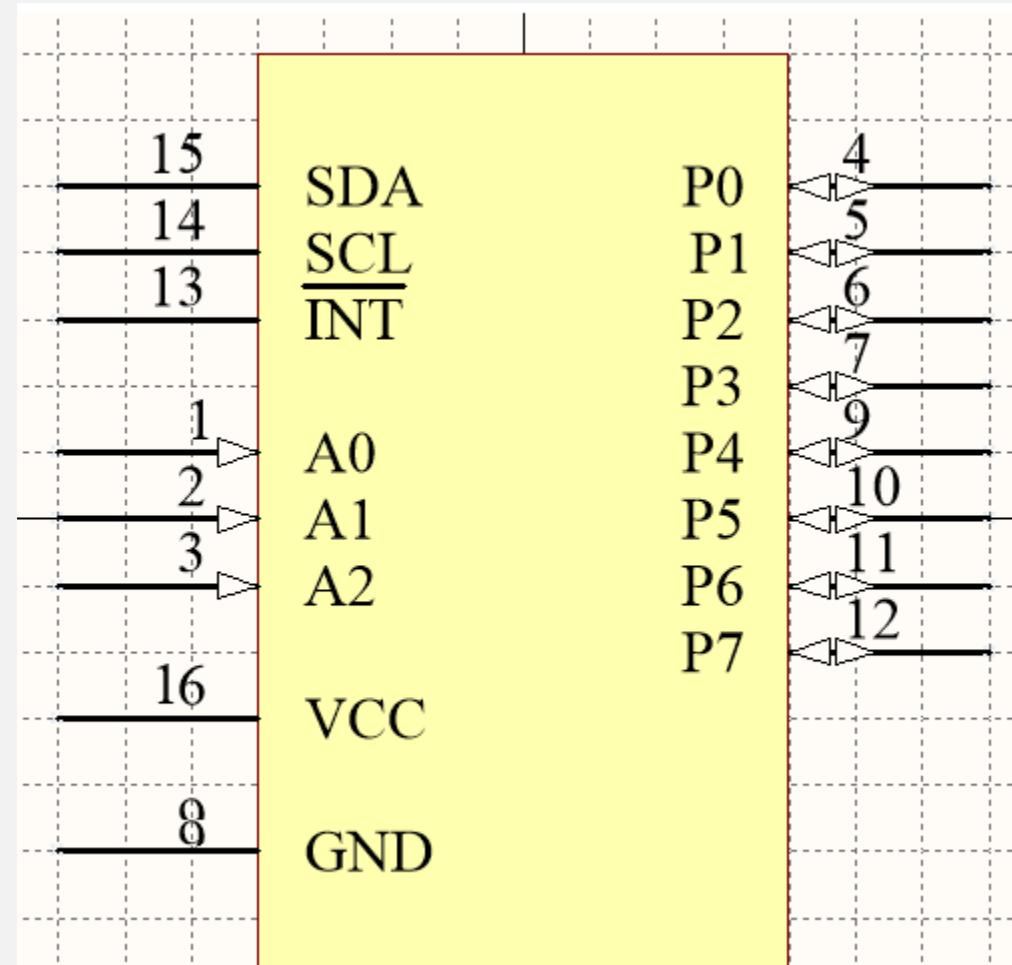
- It is a good idea to always verify that the component has the correct pinout



Pin Functions			DESCRIPTION
NO.	PIN	I/O	
1	A0	I	Address input. Connect directly to V _{CC} or ground
2	A1	I	Address input. Connect directly to V _{CC} or ground
3	A2	I	Address input. Connect directly to V _{CC} or ground
4	P0	I/O	P-port input-output. Push-pull design structure. At power on, P0 is configured as an input
5	P1	I/O	P-port input-output. Push-pull design structure. At power on, P1 is configured as an input
6	P2	I/O	P-port input-output. Push-pull design structure. At power on, P2 is configured as an input
7	P3	I/O	P-port input-output. Push-pull design structure. At power on, P3 is configured as an input
8	GND	—	Ground
9	P4	I/O	P-port input-output. Push-pull design structure. At power on, P4 is configured as an input
10	P5	I/O	P-port input-output. Push-pull design structure. At power on, P5 is configured as an input
11	P6	I/O	P-port input-output. Push-pull design structure. At power on, P6 is configured as an input
12	P7	I/O	P-port input-output. Push-pull design structure. At power on, P7 is configured as an input
13	INT	O	Interrupt output. Connect to V _{CC} through a pullup resistor
14	SCL	I/O	Serial clock bus. Connect to V _{CC} through a pullup resistor
15	SDA	I/O	Serial data bus. Connect to V _{CC} through a pullup resistor
16	VCC	—	Supply voltage

FINISHED MAKING SCHEMATIC DRAWING COMPONENT

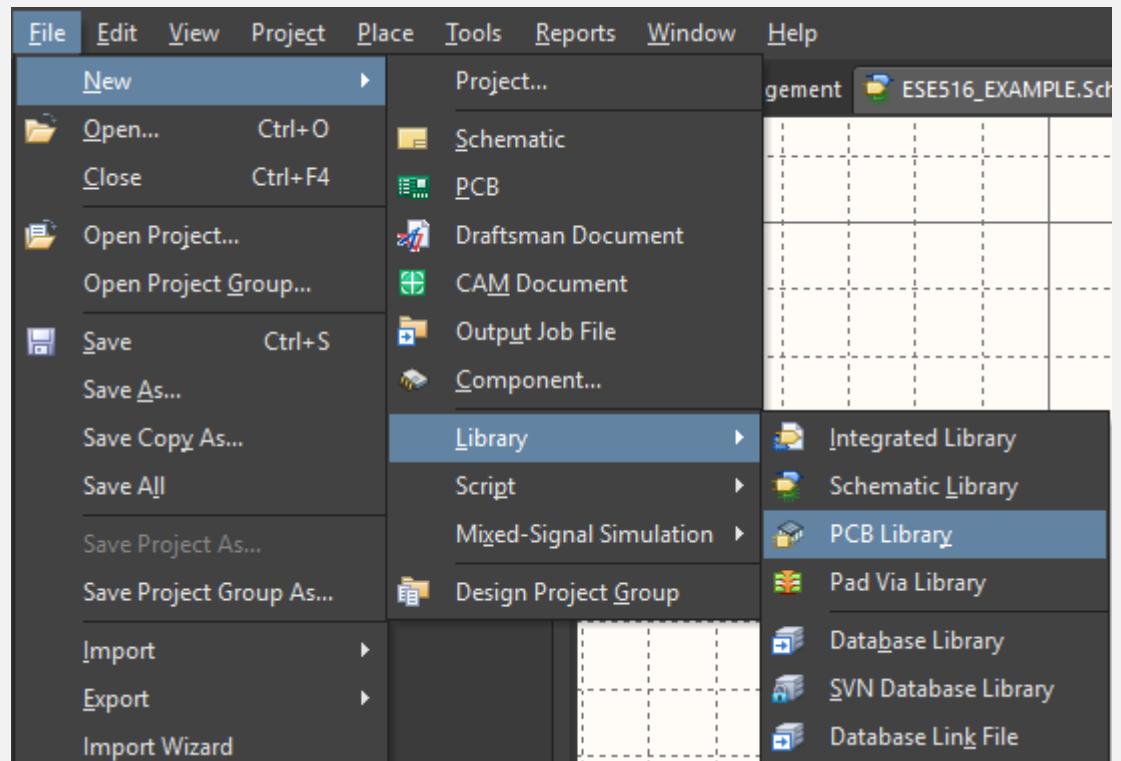
- We are done drawing the component! Now we can go to the next step – Doing the footprint. **Now is a good time to save!**



PCB LIBRARY FOR COMPONENTS

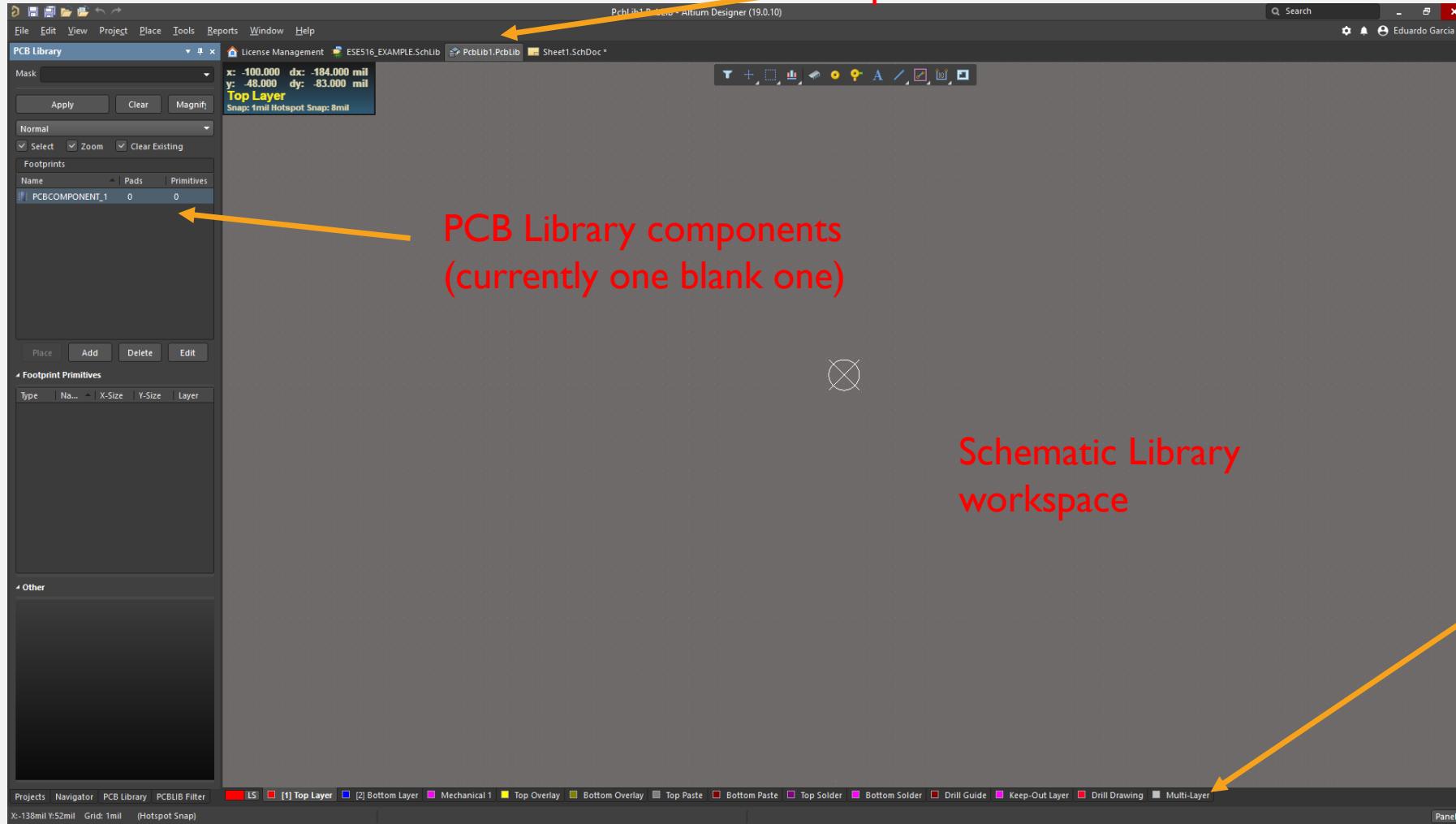
MAKING A PCB LIBRARY (FOR COMPONENTS)

- Now that we finished doing our symbol for the [TCA9534PWR](#), we now need to do the PCB footprint. This is what will actually appear on the PCB related to the component.
- To make a PCB Library, go to “File->New->Library->PCB Library”.



MAKING A PCB LIBRARY

- This will open a blank PCB Library.



TWO WAYS TO MAKE PCB COMPONENTS

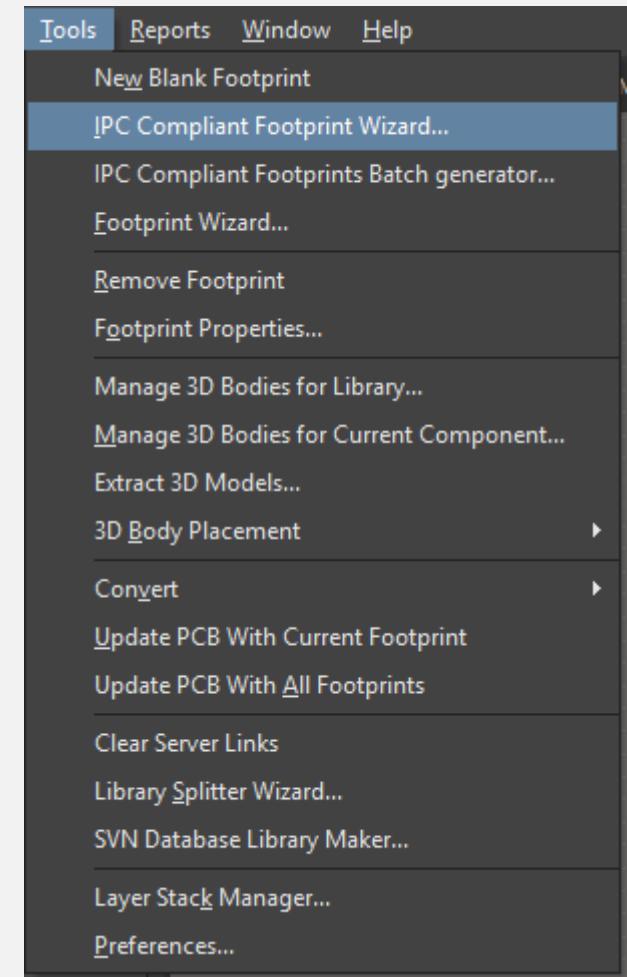
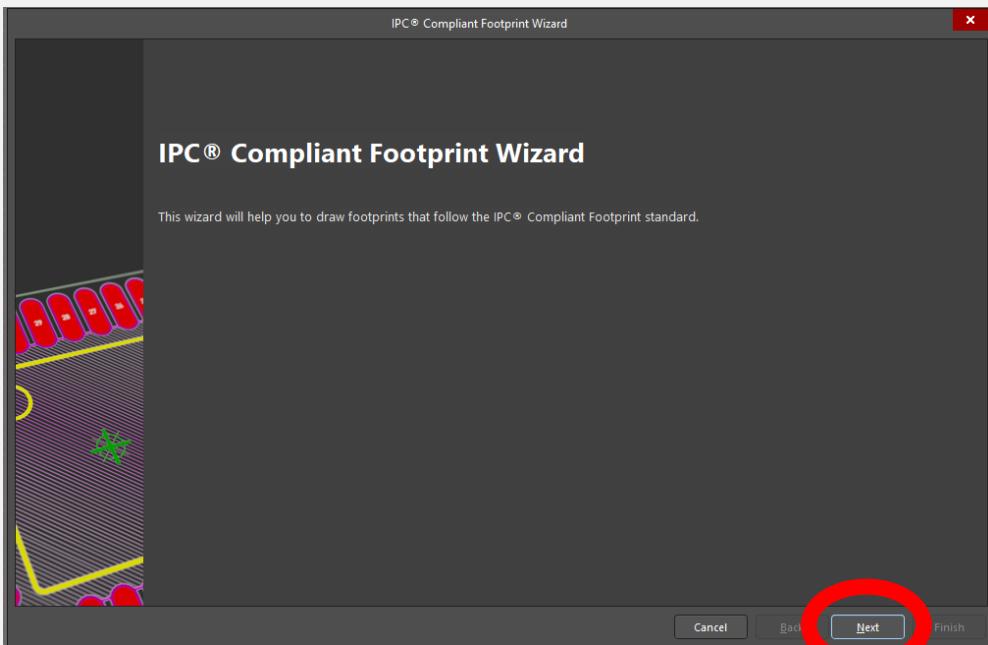
There are two main ways to do PCB footprints:

- **Manually:** You place each pad carefully, following the footprint provided by the manufacturer
 - Some manufacturers provide a “recommended footprint”, a footprint they recommend to use for their component. Fortunately, our component ([TCA9534PWR](#)) has this on PG: 36 of the datasheet (Land Pattern Example)
 - When there is no recommended footprint on the datasheet, check if you can do it with Altium’s IPC Compliant Footprint Wizard. If not, you are out of luck! You will need to do it from scratch or search online for an example!
- **Using Altium’s IPC Compliant Footprint Wizard:** Altium can auto-generate a component footprint given the component’s dimensions. This is only possible for components with common footprints (such as DFN or SOIC). If the Wizard cannot make it for you, you must do it manually!

We will see both ways in this tutorial

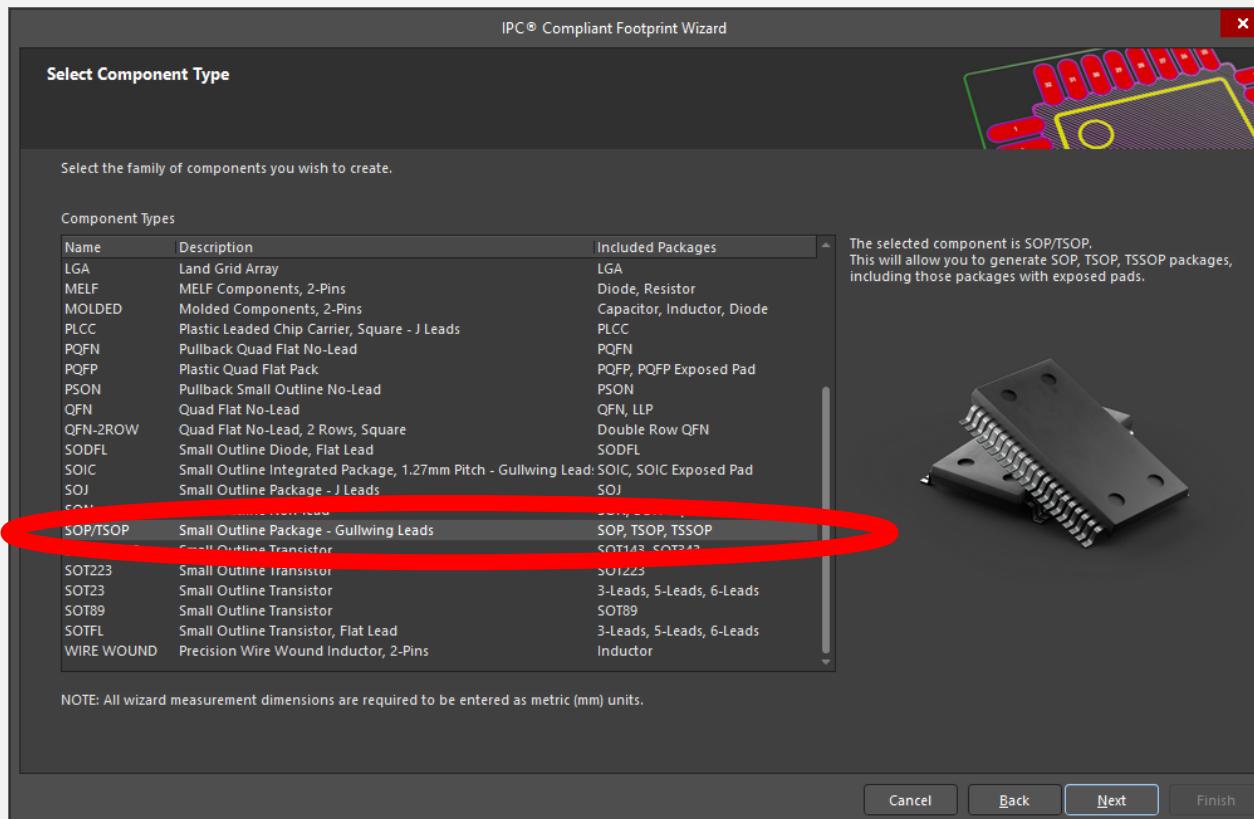
MAKING A COMPONENT USING ALTIUM'S IPC LIBRARY

- Let's start with the easiest way – suing Altium's IPC Component Library. Go to “Tools -> IPC Compliant Footprint Wizard...”
- The IPC Compliant Wizard window will appear – Hit Next.



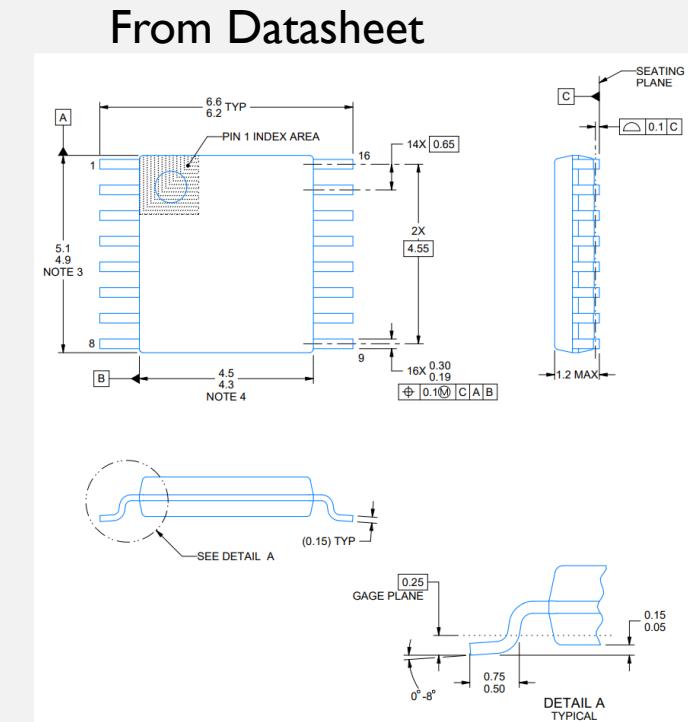
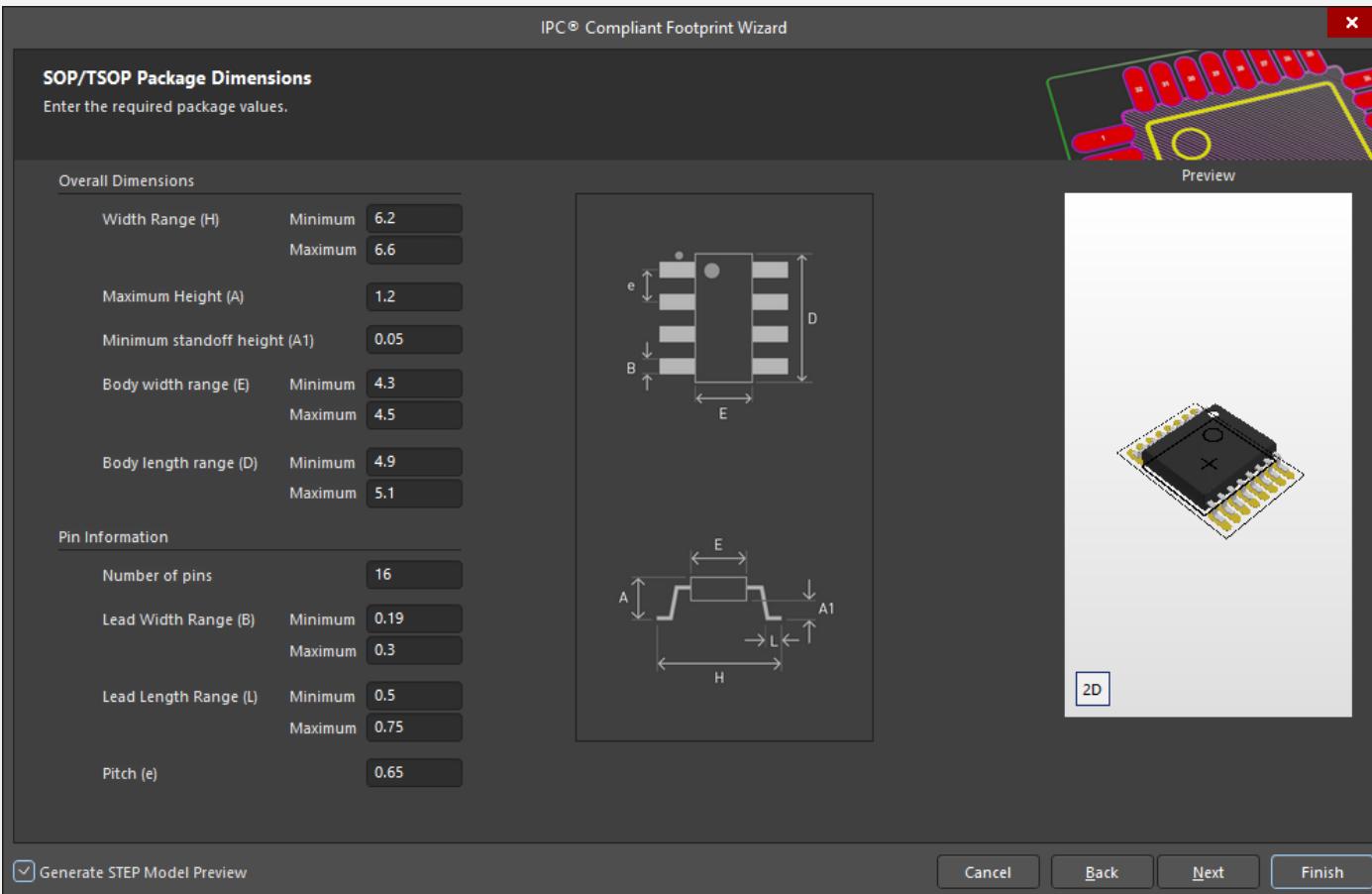
MAKING A COMPONENT USING ALTIUM'S IPC LIBRARY

- Here you will see the type of packages Altium's IPC Compliant wizard can handle. Any type of footprint that is not on this list, Altium cannot make it with this wizard and it must be made by hand. Thankfully Altium has the package type we need – SOP/TSOP (Very similar to the package we need - 16-TSSOP). Choose that and hit Next.



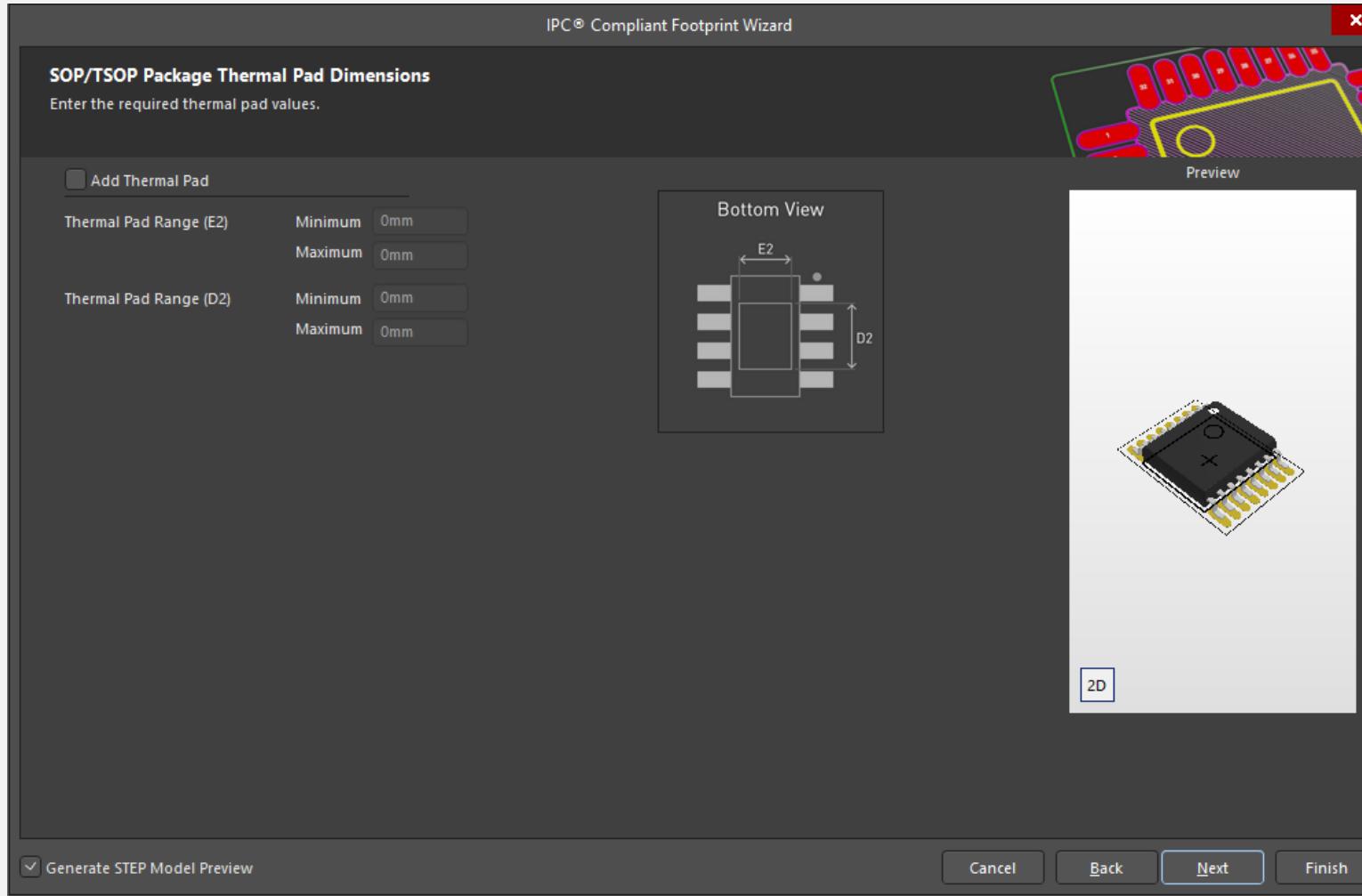
MAKING A COMPONENT USING ALTIUM'S IPC LIBRARY

- The IPC wizard works by using rules to make the footprint based on the geometry of the device. You will need to find the geometry of the device on the datasheet and fill the information Altium is asking of you. The mechanical drawing of the component is found on PG: 35 of the datasheet. Fill the information on this table.



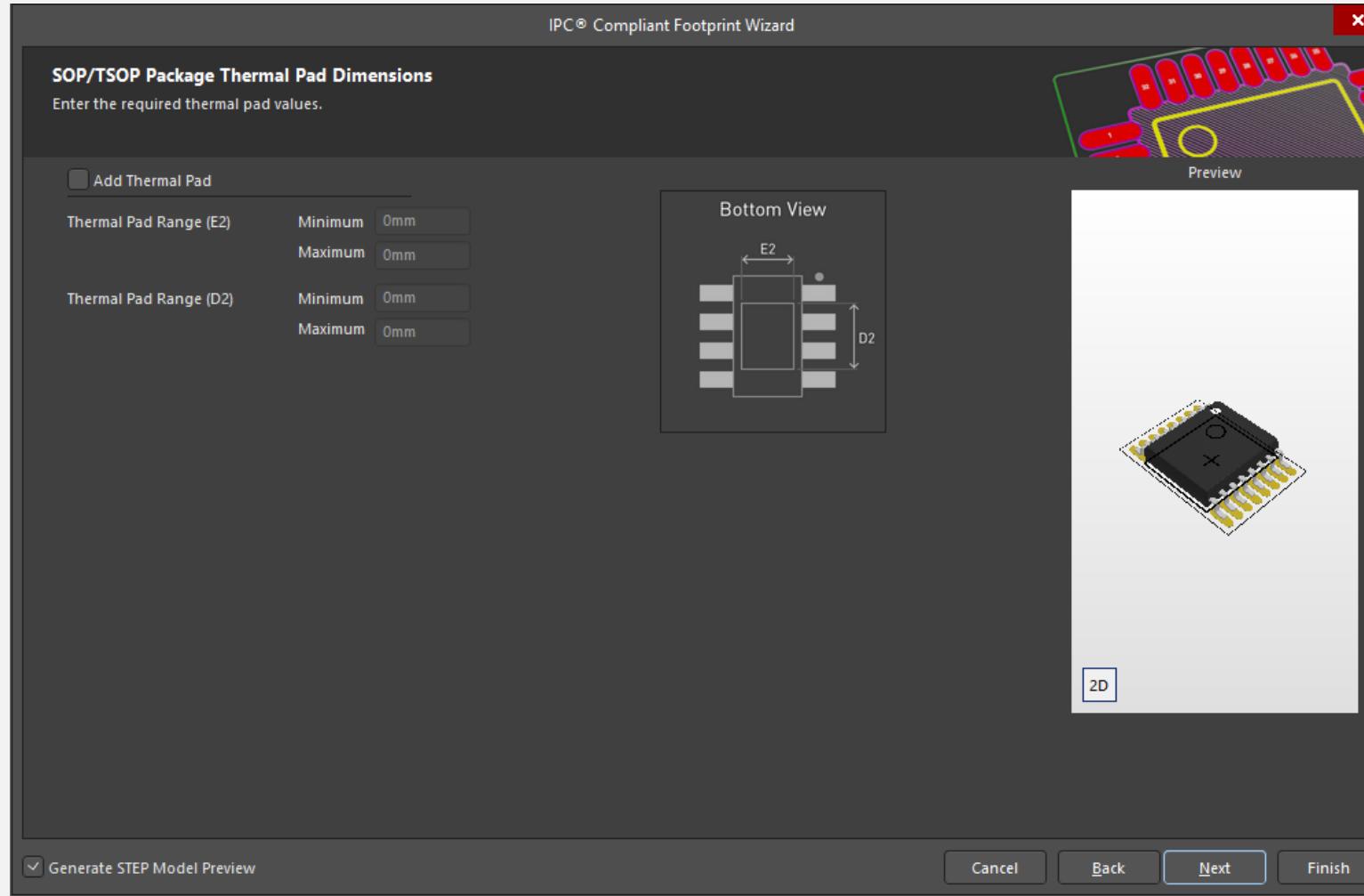
MAKING A COMPONENT USING ALTIUM'S IPC LIBRARY

- This device has no thermal pad, so hit Next.



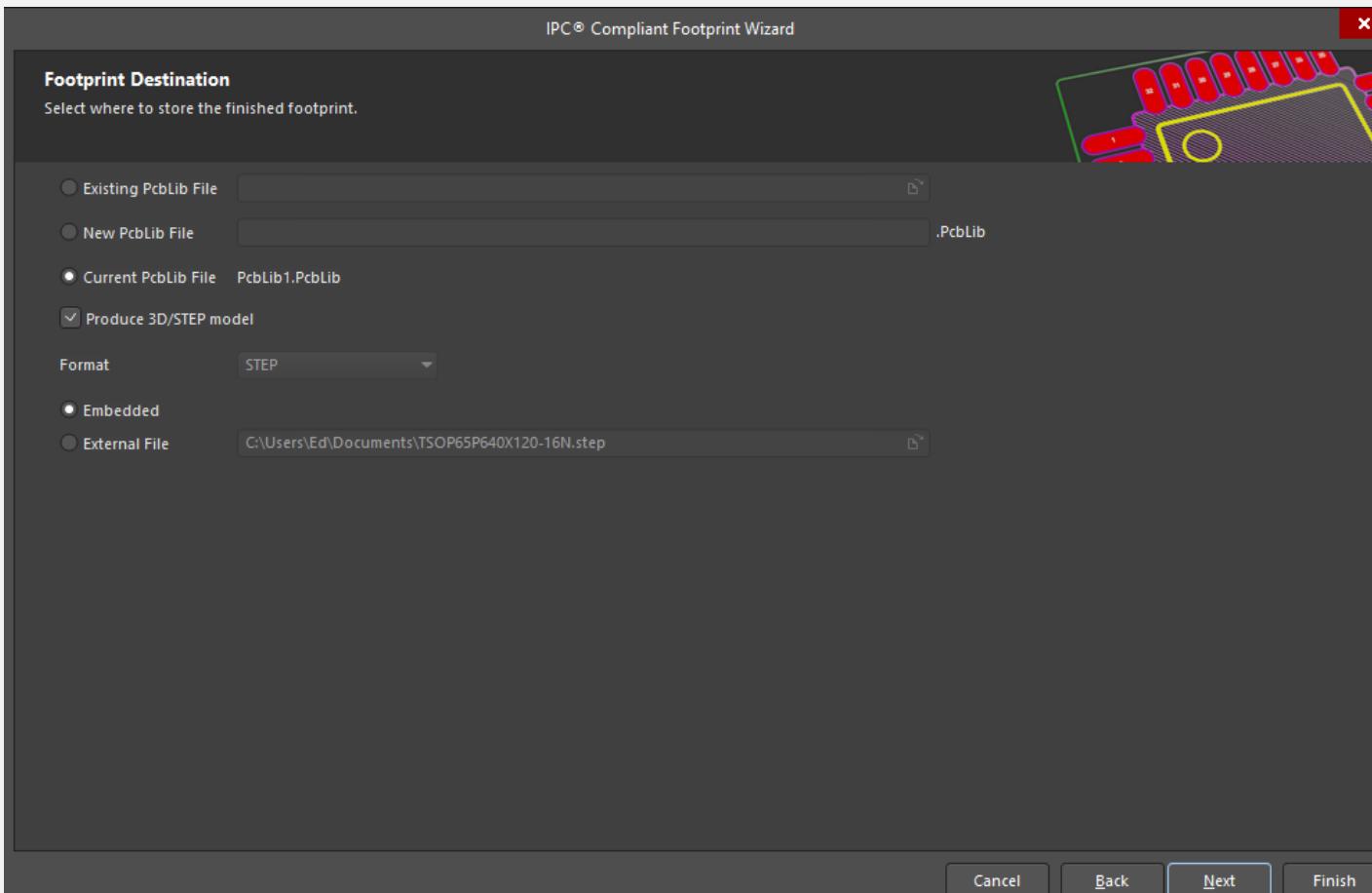
MAKING A COMPONENT USING ALTIUM'S IPC LIBRARY

- This device has no thermal pad, so hit Next.



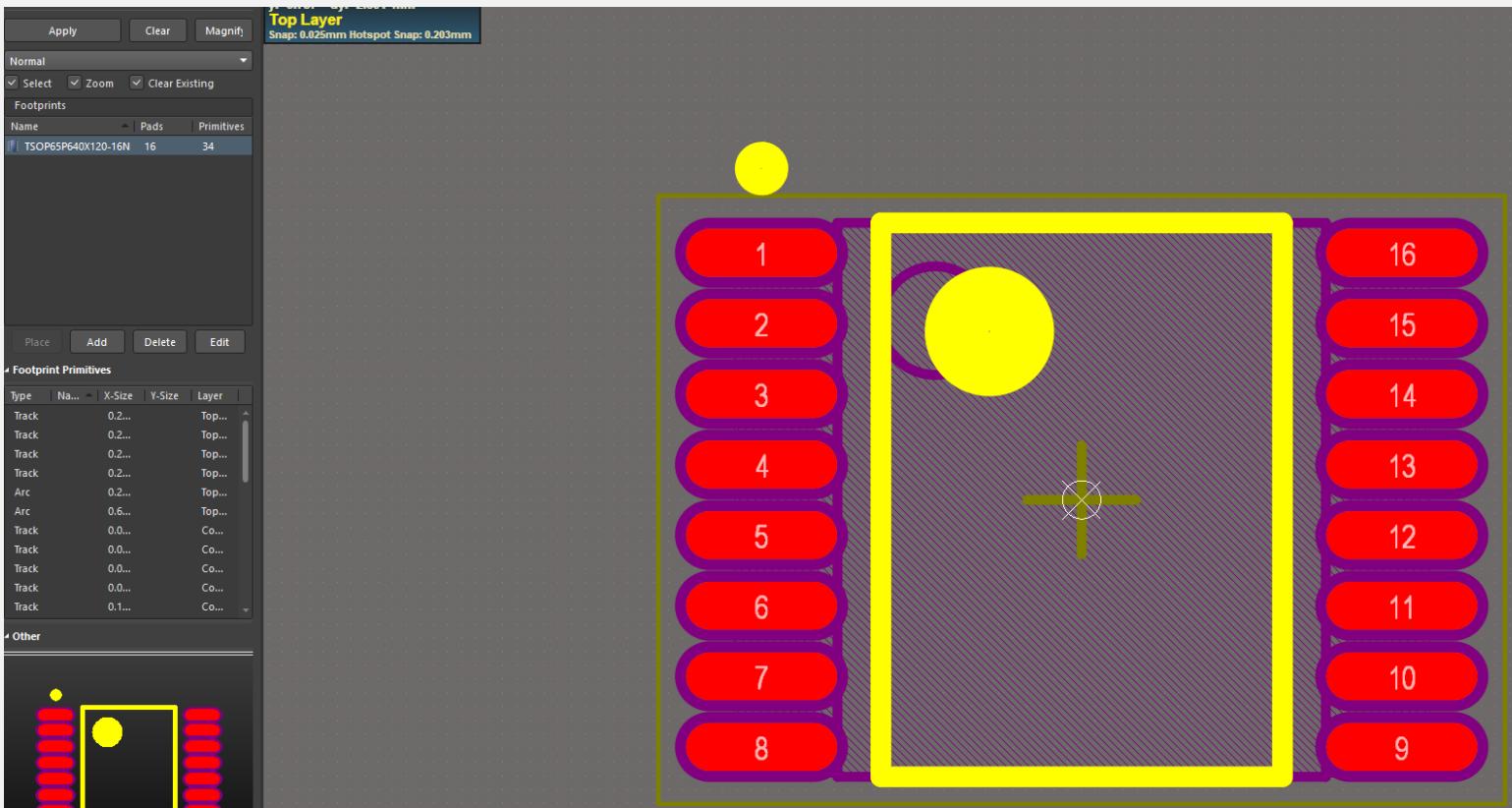
MAKING A COMPONENT USING ALTIUM'S IPC LIBRARY

- As you hit next, the rest of the options are OK. Please hit next until you get to the following screen. Check that you have the option “Current PcbLib file” and that “Produce 3D/STEP model” is checked. Hit Next and Finished.



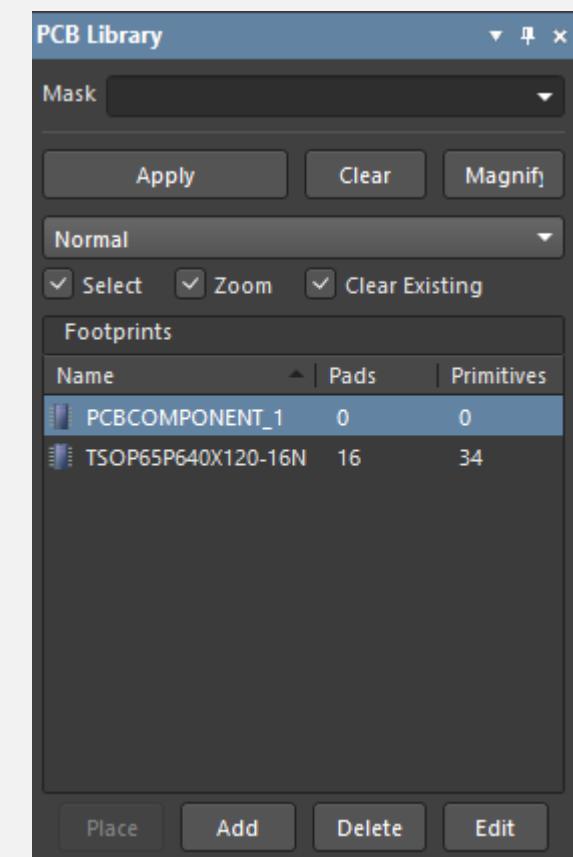
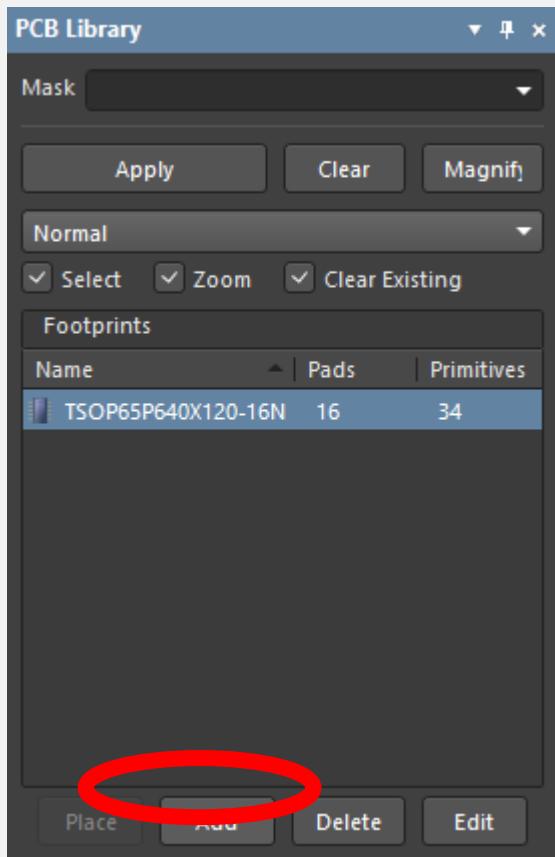
MAKING A COMPONENT USING ALTIUM'S IPC LIBRARY

- Congratulations! You have your component! Altium PCB IPC compliant Wizard is the easiest way to make a footprint (If it can make that footprint). To learn how to do it by hand, we will make another PCB Library component on the next steps!



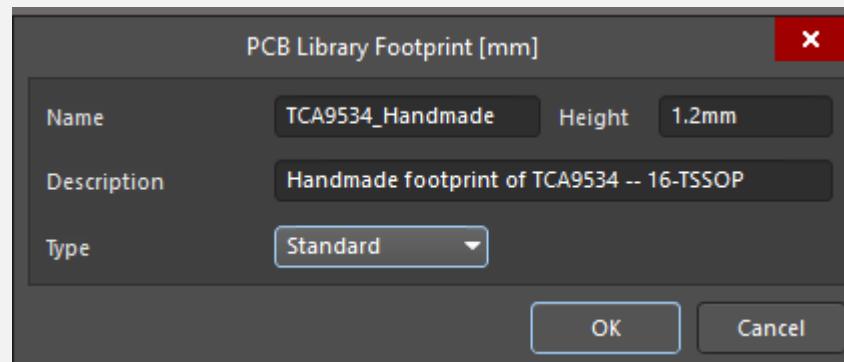
MAKING A PCB COMPONENT BY HAND

- Let's start a new component to learn how to do it by hand. Make a new component by hitting the "Add" key on the "PCB Library" panel. This will add a new component called "PCBCOMPONENT_1".



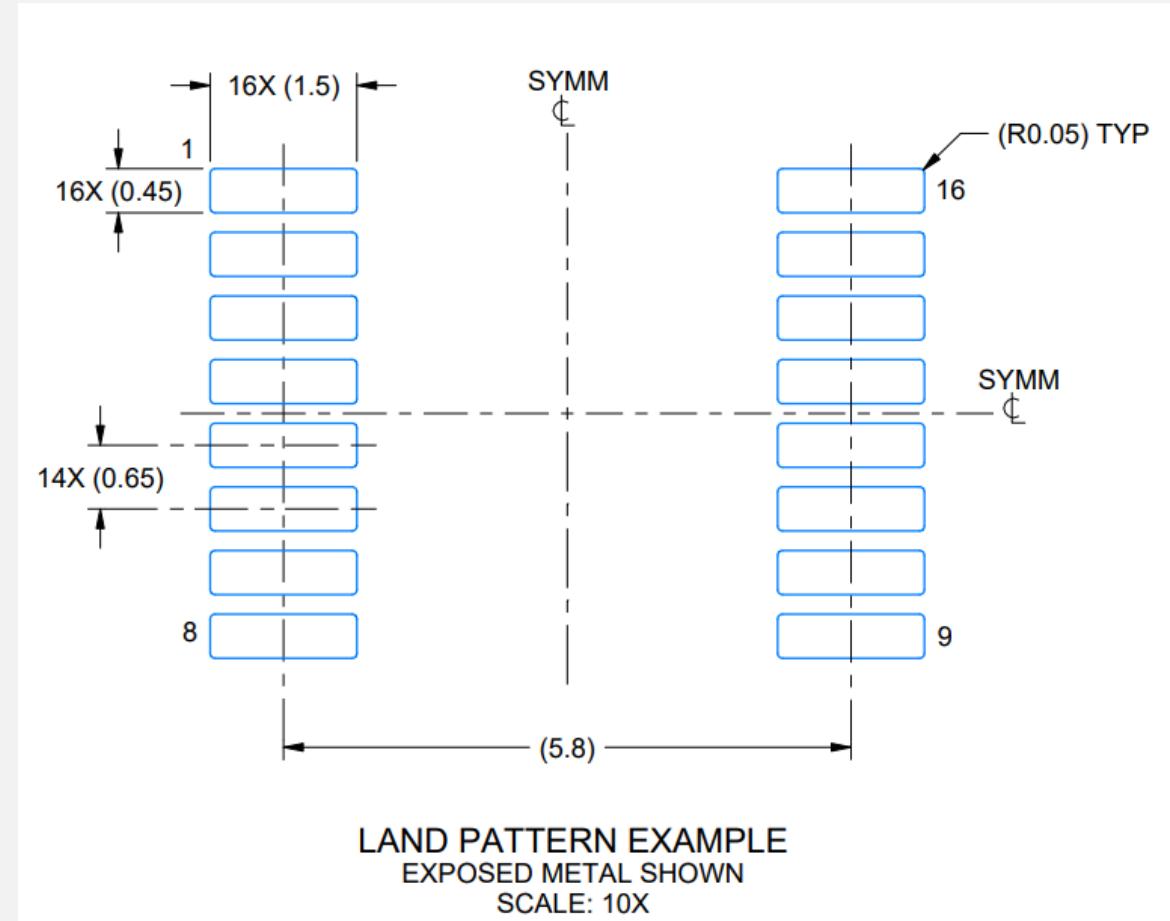
MAKING A PCB COMPONENT BY HAND

- Double click on “PCBCOMPONENT_I”. Change the name to the following.
- **It is a good practice to make unique PCB Names for each component. It can lead to really weird bugs if you have components sharing name!**
- **Also good practice – to add the “height” of the component. I put the max height of the component that you find on the Datasheet. When you are doing your PCBs, you can add height restriction to areas of your PCB, and Altium will use this data to know how high a component is.**



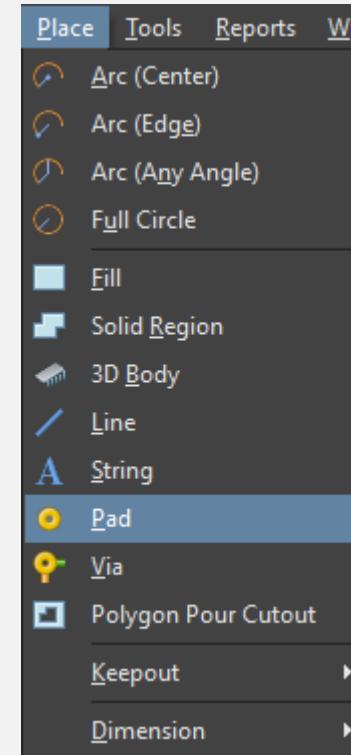
MAKING A PCB COMPONENT BY HAND

- For this component, we will follow the recommendation on page 36 of the datasheet.
- Good practices:
 - Put the center coordinates (0,0) at the center of the component. Some people do the center of Pin 1 – Either is fine as long as it is consistent
 - This coordinate is used by the pick and place machine to place the component on the PCB. Because of this, I believe putting (0,0) at the center of the component is better.



MAKING A PCB COMPONENT – PLACING PADS

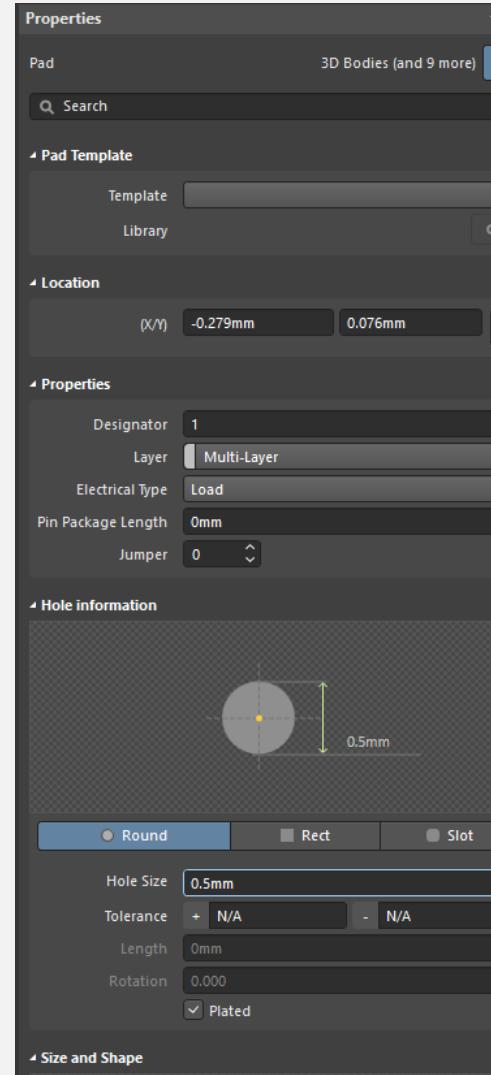
- The first step is to place the pads, sixteen of them, into the PCB.
- To place a pad, go to “Place -> Pad”. You can also press “p” then “p” again
 - “p” -> shortcut for Place
 - “p” -> shortcut for Pad



A pad will now appear attached to your cursor. You can place it on the board to edit its properties with a double click, or you can press TAB while it is attached to your cursor to edit its properties. I prefer this way.

MAKING A PCB COMPONENT – PLACING PADS

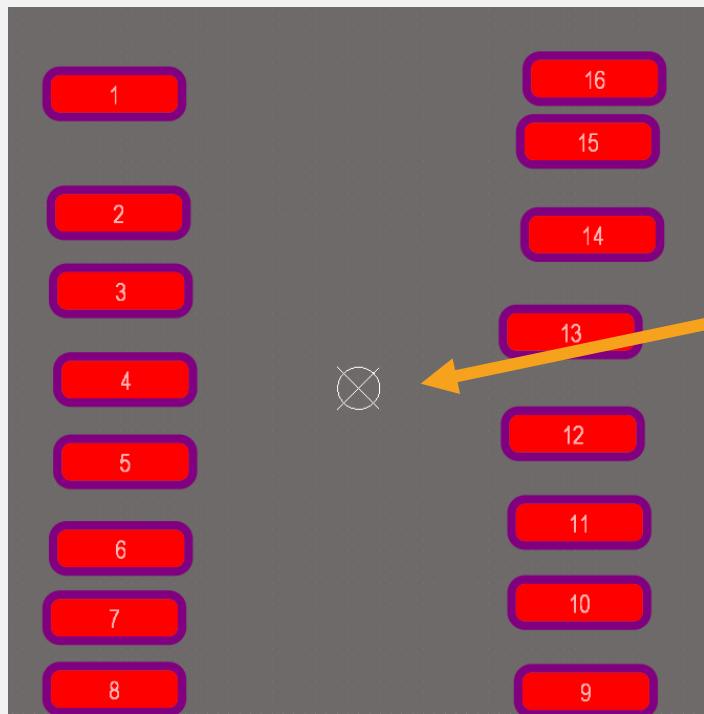
- Press TAB to change the properties of the Pad. There are a lot of options here to discuss:
- **Properties**
 - **Designator:** Will be Pin #. Will relate to the Pin# of the Schematic symbol – Pin 1 of the schematic symbol will connect to pin 1 of the PCB library.
 - **Layer:** Determines the copper layer the pad is. For Surface Mount Components like this one, we use the TOP LAYER. If you need to do thru-hole components, use MULTI LAYER
 - **Electrical Type, Pin Package Length and Jumper** are special properties out of scope of this tutorial



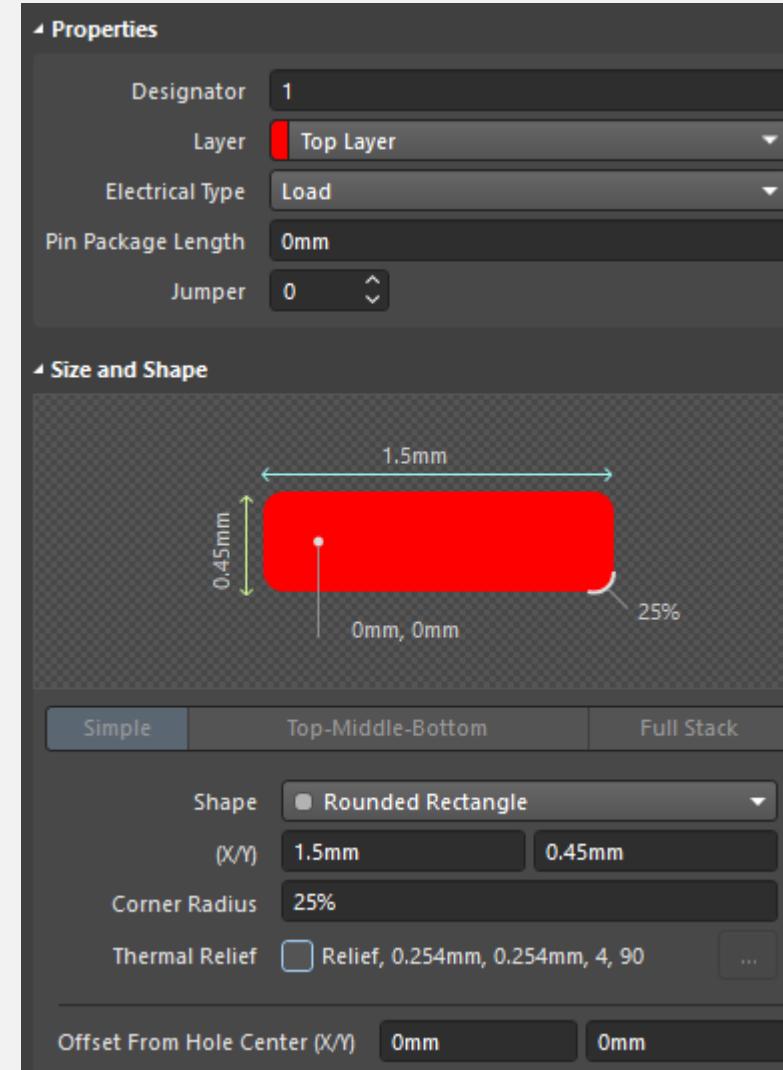
- **Hole Information** only appears if you select the layer to be MULTI-LAYER. This is where you put the size of the hole for through-hole components.
- **Size and Shape** determines the shape of your pad. You can do circular pads, rectangular, rounded rectangles, etc.
- **Solder mask, paste mask and Test point** are out of the scope for this tutorial

MAKING A PCB COMPONENT – PLACING PADS

- Choose the following for your PAD and place 16 in the shape of the part. We will change the coordinates of the pads later on – we just want to place them for now. Altium will auto-increment the PIN number as you place pads.

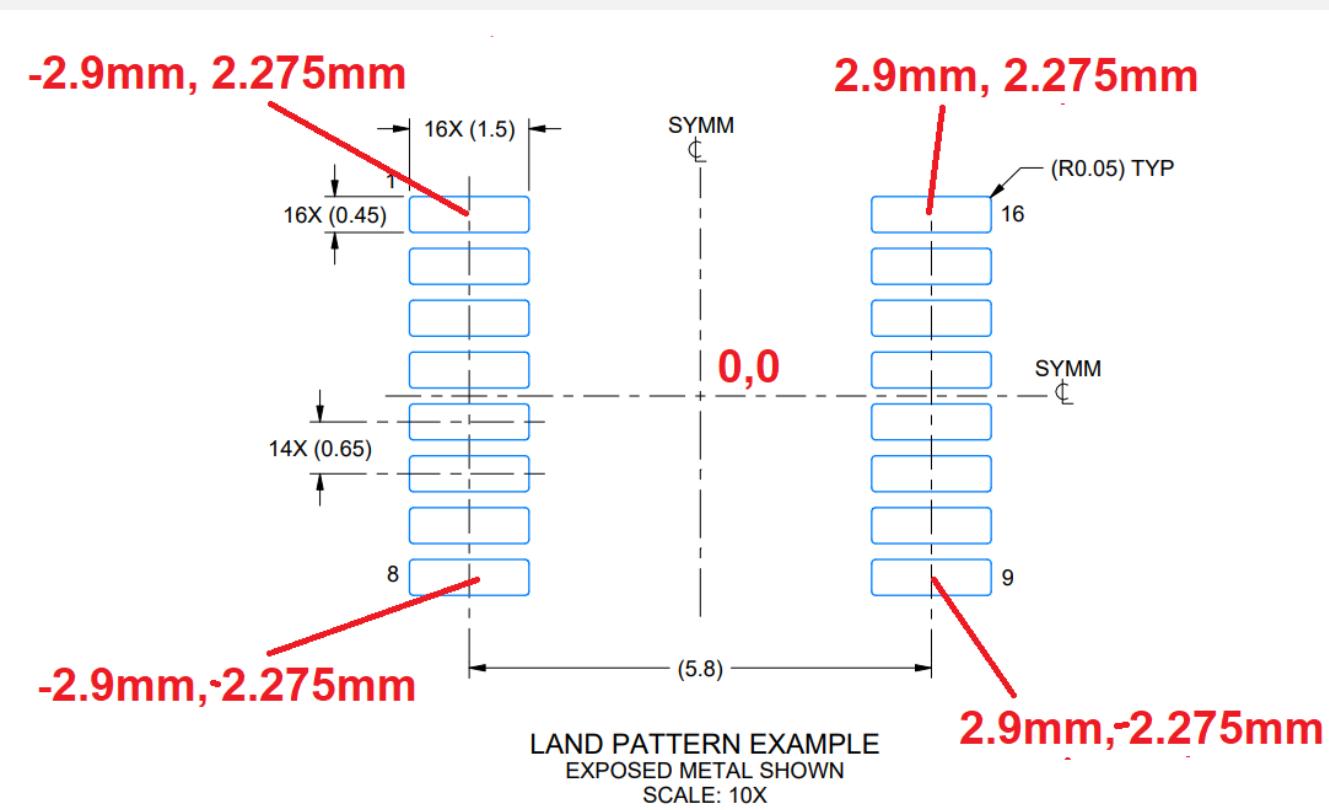


Part Center
(0,0)
(Do control + End to find it, or press “j”



MAKING A PCB COMPONENT – PLACING PADS

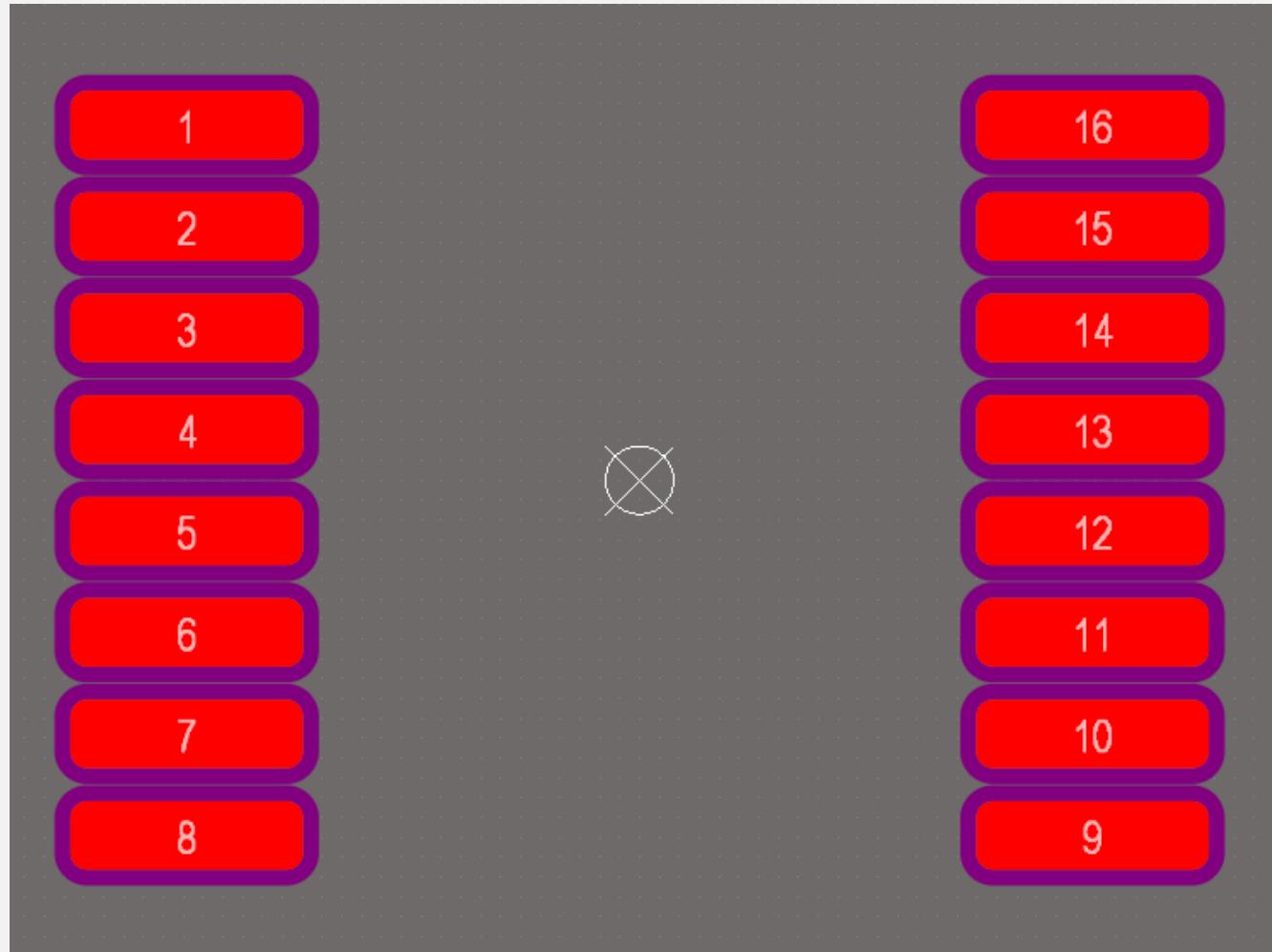
- Now that we have 16 pads on the board, we have to put the X/Y coordinates for each one. Unfortunately, we have to do some math to take the X/Y coordinates of each Pad from the Datasheet. This is very common to do!



MAKING A PCB COMPONENT – PLACING PADS

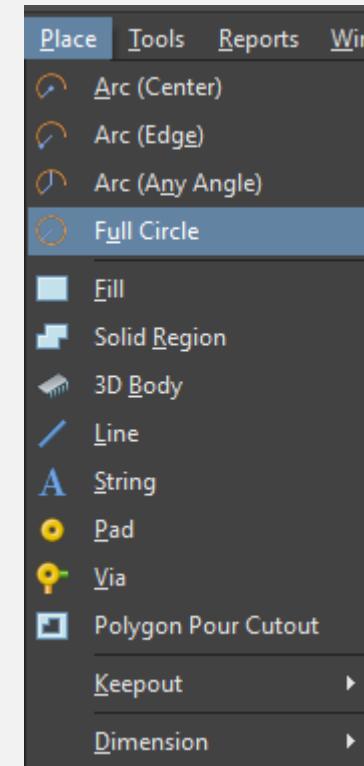
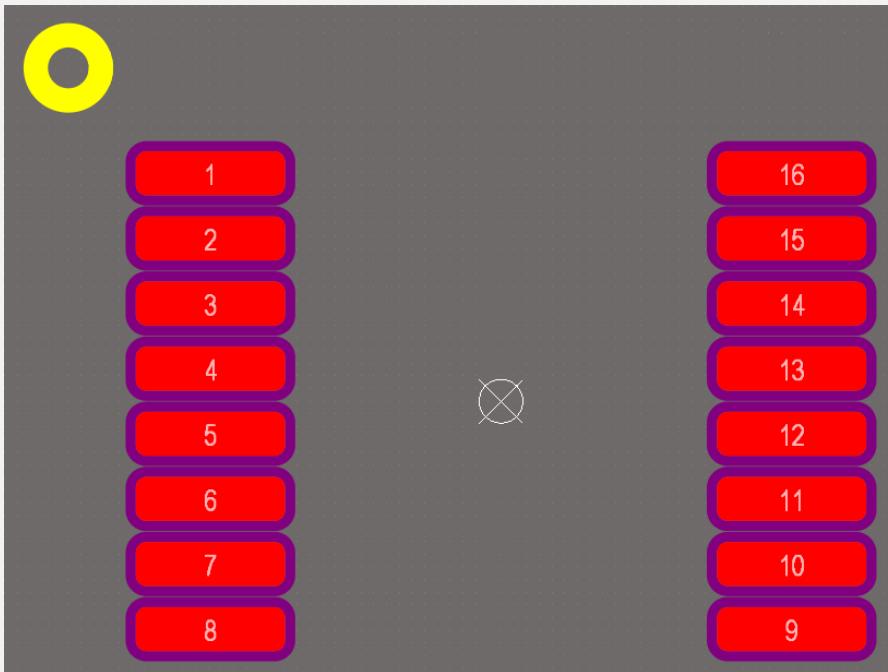
- Change the X,Y coordinates of the pads in the following way:

PIN NUMBER	X (mm)	Y (mm)
1	-2.9	2.275
2	-2.9	1.625
3	-2.9	0.975
4	-2.9	0.325
5	-2.9	-0.325
6	-2.9	-0.975
7	-2.9	-1.625
8	-2.9	-2.275
9	2.9	-2.275
10	2.9	-1.625
11	2.9	-0.975
12	2.9	-0.325
13	2.9	0.325
14	2.9	0.975
15	2.9	1.625
16	2.9	2.275

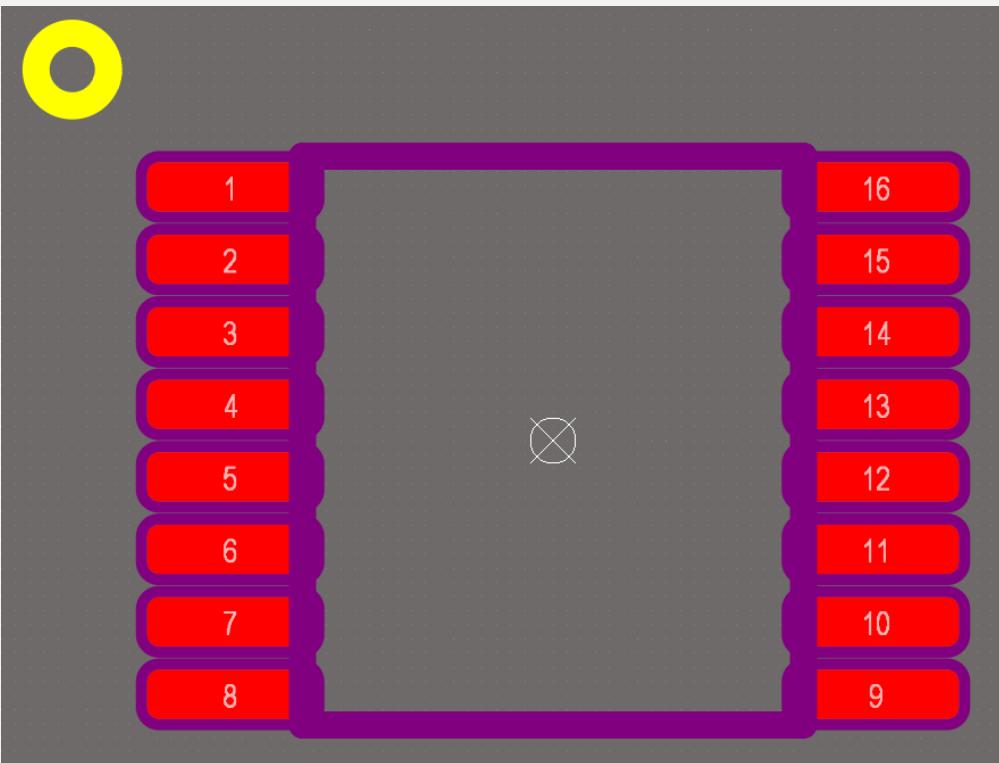


MAKING A PCB COMPONENT – PLACING PADS

- We have the pads now – now we have to add the Silkscreen indicator for Pin#1 – A crucial detail!
- Go to “Place ->Full Circle” and place a circle near Pin1. This is used to indicate on the PCB which pin is Pin 1. It helps developers as well as the board fabrication house.
- Change the layer of the circle to “Top Overlay”

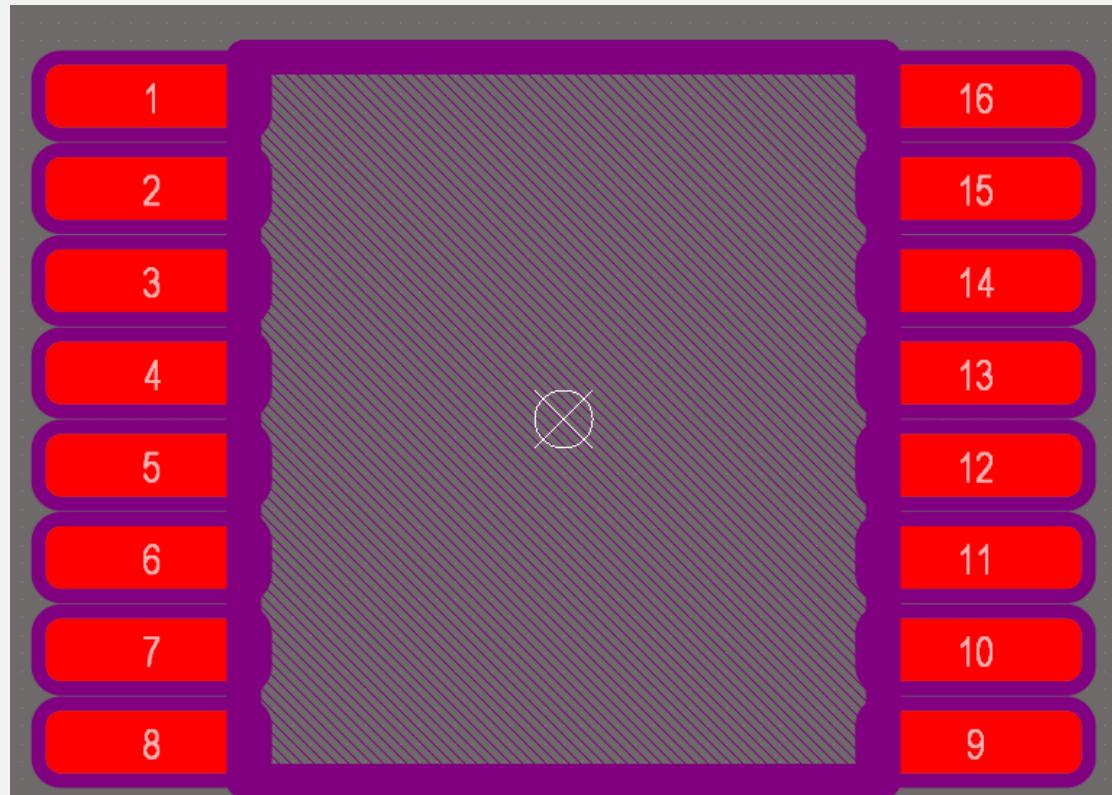


OPTIONAL – ADDING A COARSE 3D MODEL

- **Optional:** Let's add now a very coarse 3D model. We need to draw some lines to help us in the next steps. To draw the lines, go to “Place ->Line”
 - We are going to draw the outline of the physical circuit (it is 4.5mmx5.1mm). Draw a line with the following coordinates:
 - Top Left - -2.25mm,2.55mm
 - Top Right: 2.25mm, 2.55mm
 - Bottom Left - -2.25mm,-2.55mm
 - Bottom Right: 2.25mm, -2.55mm
 - To easily jump to a coordinate, press “j” and then “New Location”
 - Tip: If you have a step file, you can add it also!
- 

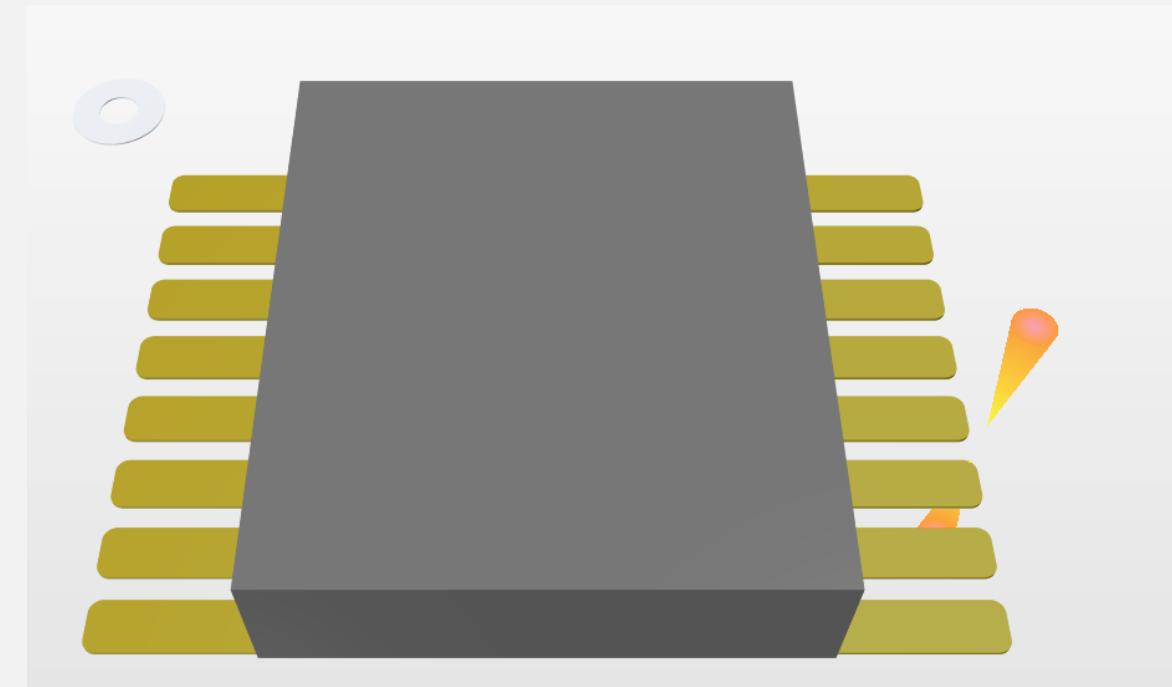
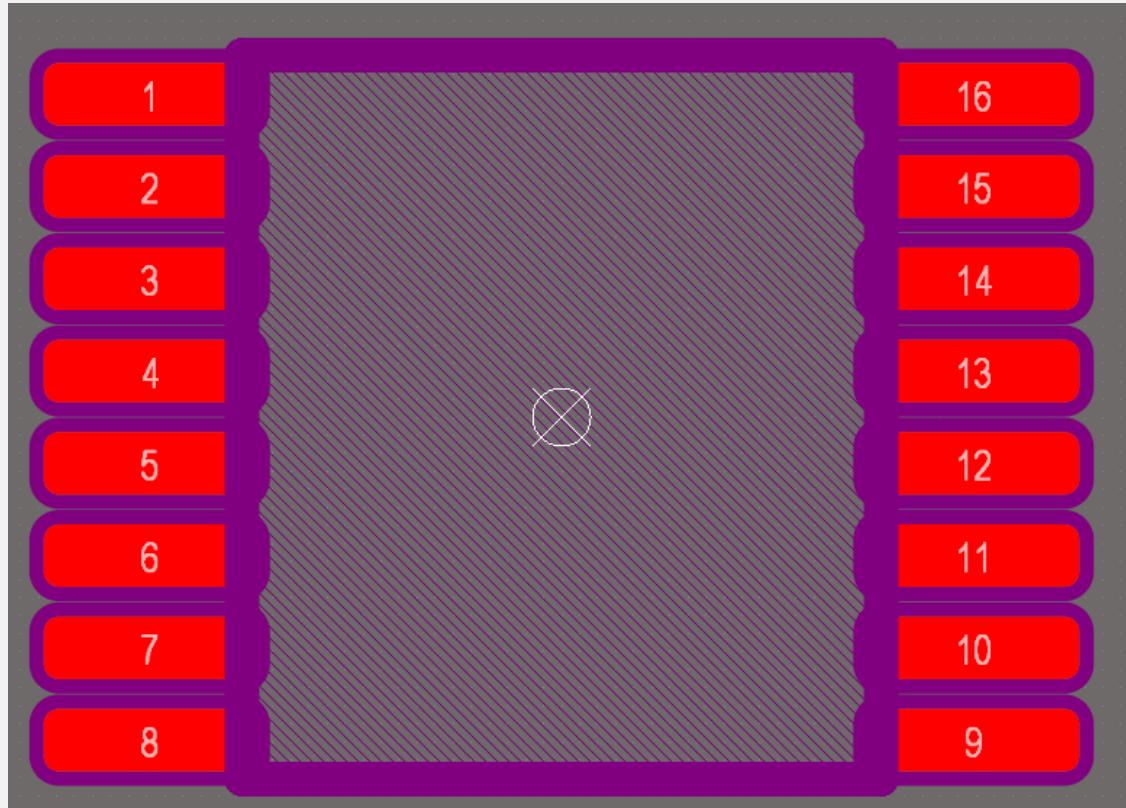
OPTIONAL – ADDING A COARSE 3D MODEL

- **Optional:** Note that we have an outline, it will be much easier to draw. Select “Place -> 3D Body”. A cursor will appear. It will snap to the line you just drew – Draw a rectangle using the last step as a guide. When done, press ESC. Give it a height of 1,2mm.



OPTIONAL – ADDING A COARSE 3D MODEL

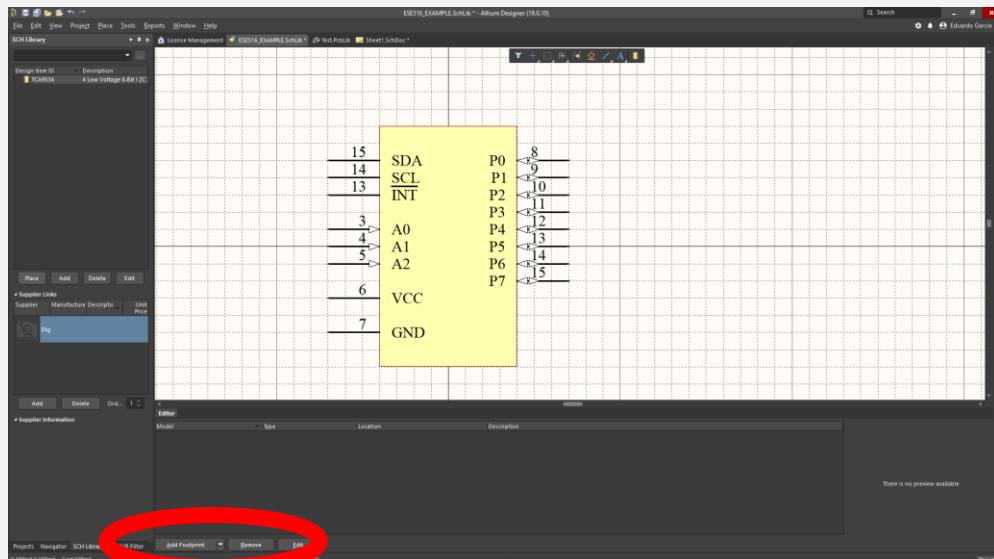
- **Optional:** You can see the PCB model in 3D by pressing '3' on the keyboard (3D View). You can return to the 2D view by pressing '2'.



LINKING THE PCB MODEL AND THE SCHEMATIC MODEL

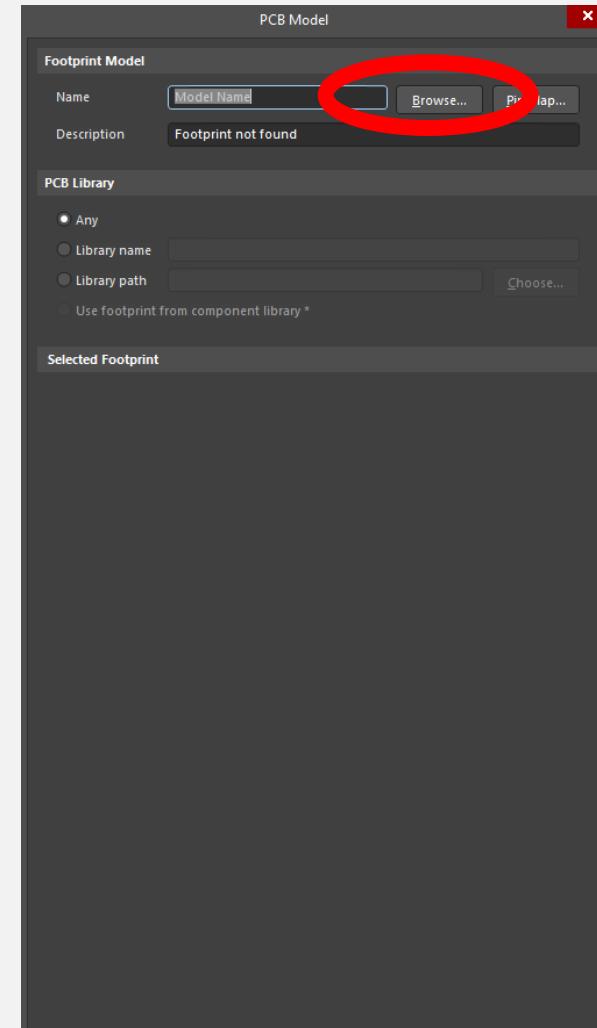
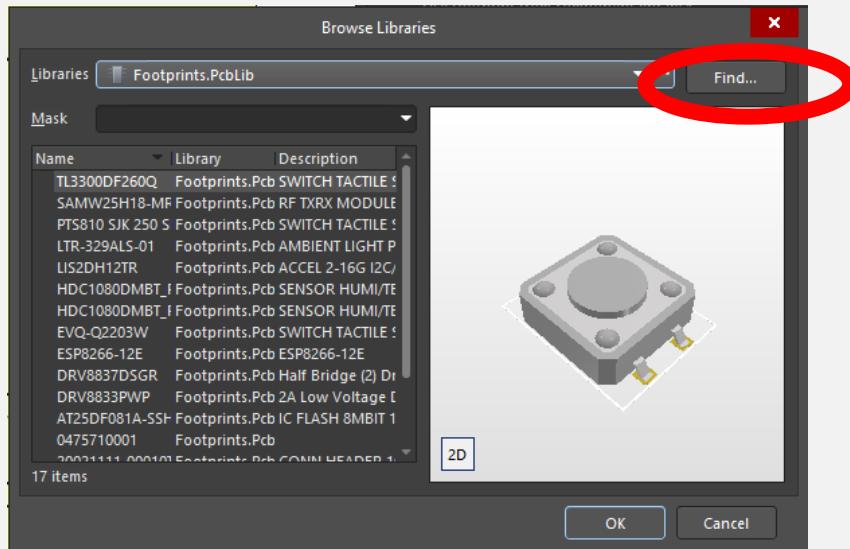
Now that we have a Schematic model and a PCB library model, we can link them together, so Altium knows that the symbol library we made must also use the PCB footprint we made. For this example we will use the footprint made by Altium, since it has more detail than the one we made by hand.

- Open the Schematic Library where we did the symbol. Choose the symbol, and on the bottom panel called “Editor” click on “Add Footprint”



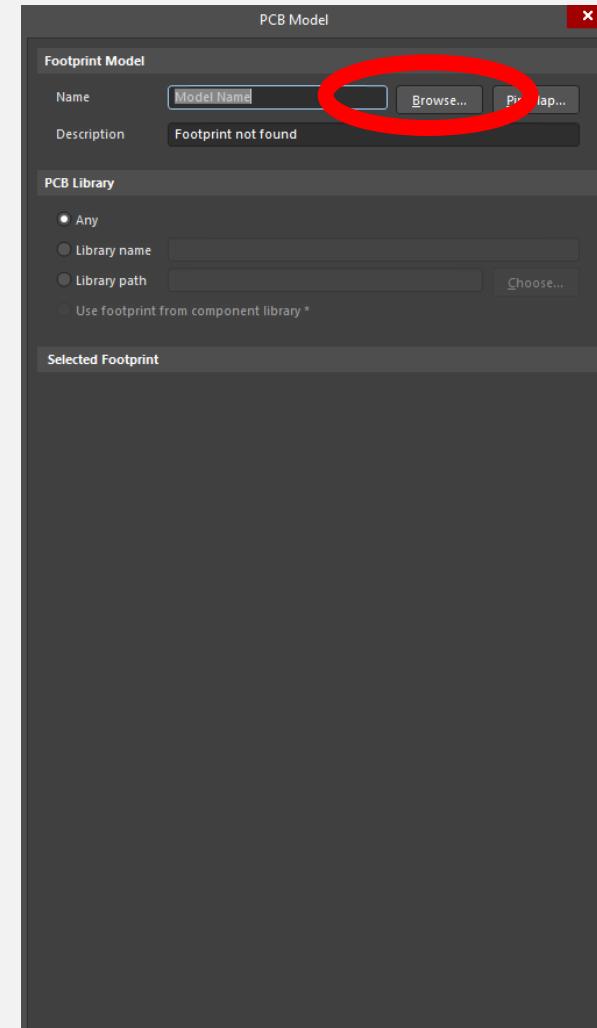
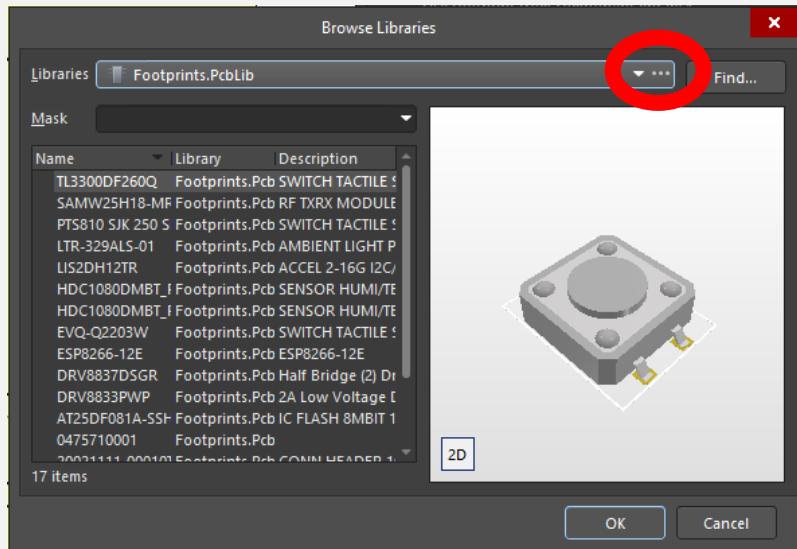
LINKING THE PCB MODEL AND THE SCHEMATIC MODEL

- We will have to tell Altium where our PCB library is and the path where to find it. Hit “Browse”..
- Most probably our Library will not be in the following window’s Library’s dropdown. Click “Find...”



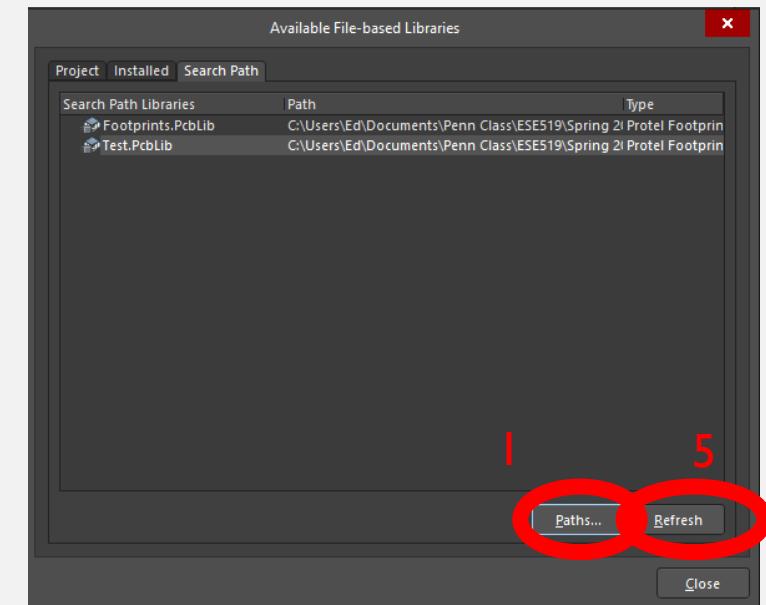
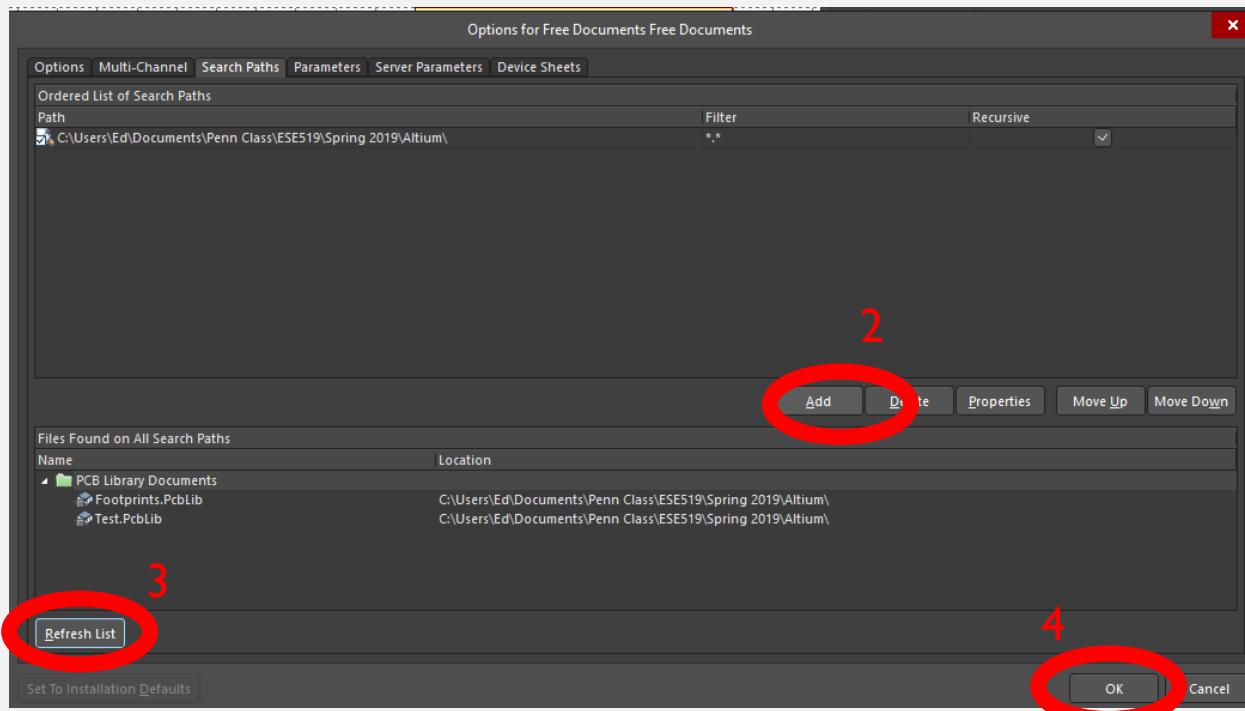
LINKING THE PCB MODEL AND THE SCHEMATIC MODEL

- We will have to tell Altium where our PCB library is and the path where to find it. Hit “Browse”..
- Most probably our Library will not be in the following window’s Library’s dropdown. Click “...”



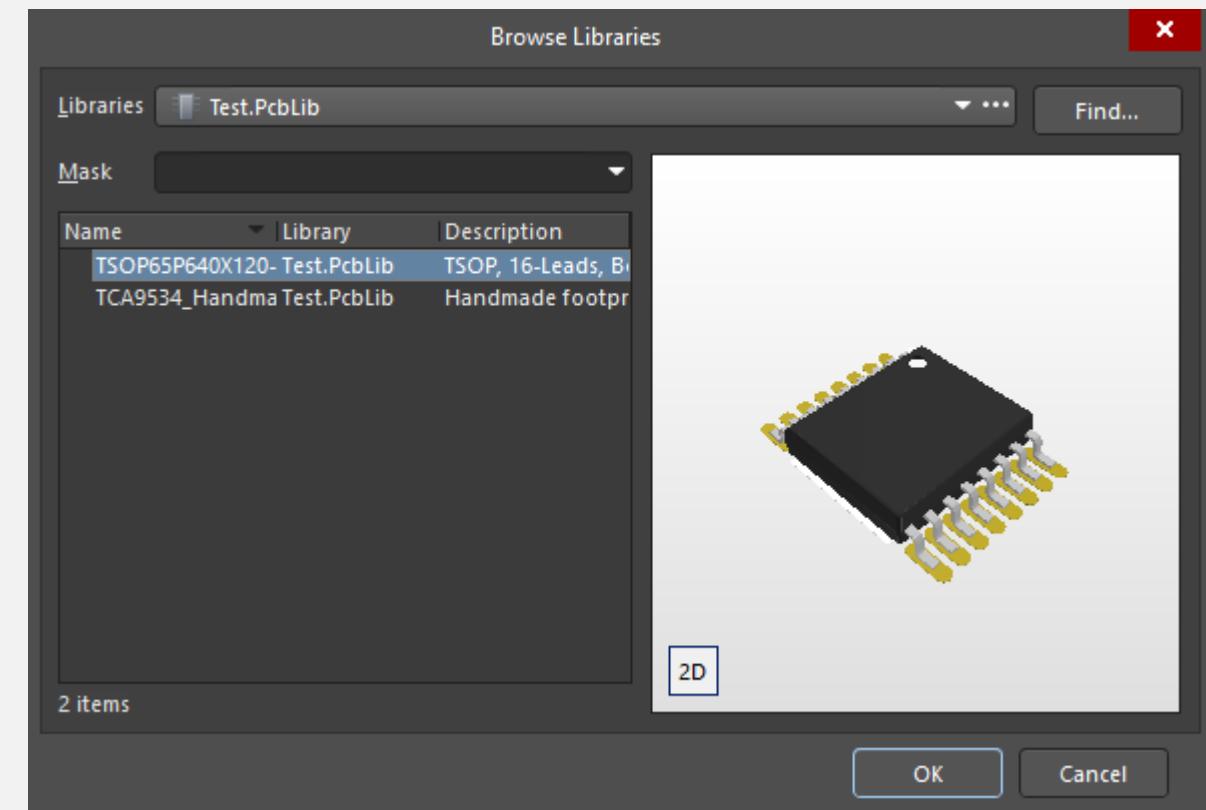
LINKING THE PCB MODEL AND THE SCHEMATIC MODEL

- Click “Paths...” on the new window, then “add” and add the path to your PCB Library.
- Once done, hit “Refresh List”, and Ok. Hit Refresh on the previous window too and hit Close



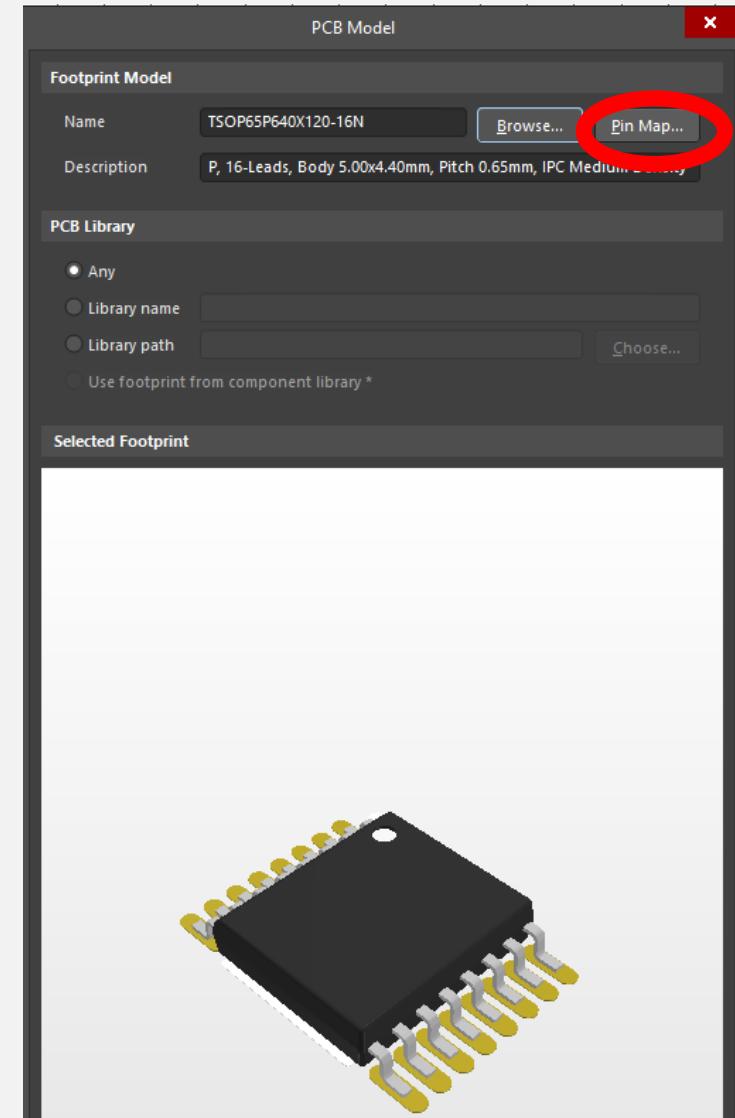
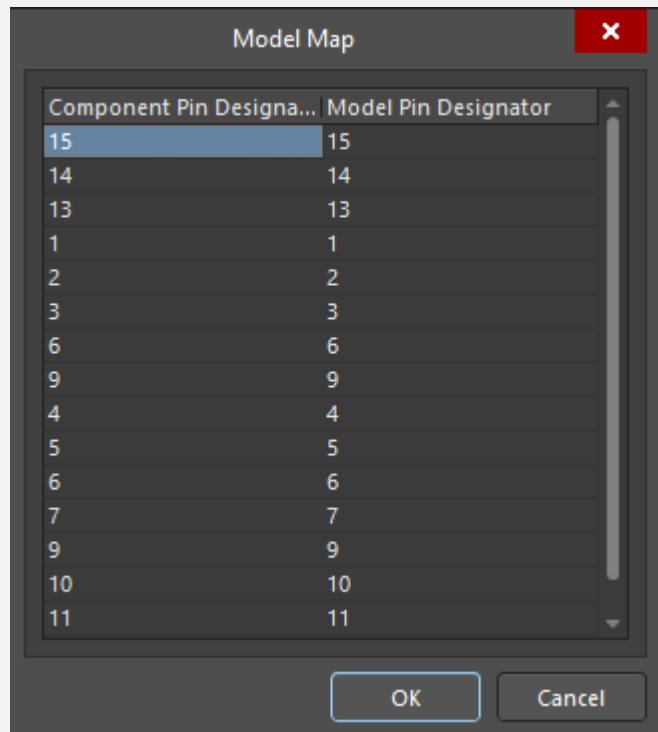
LINKING THE PCB MODEL AND THE SCHEMATIC MODEL

- Now, your library should appear on the dropdown. Select your PCB Library and select the PCB Component we made with the IPC Compliant Wizard. Hit OK.

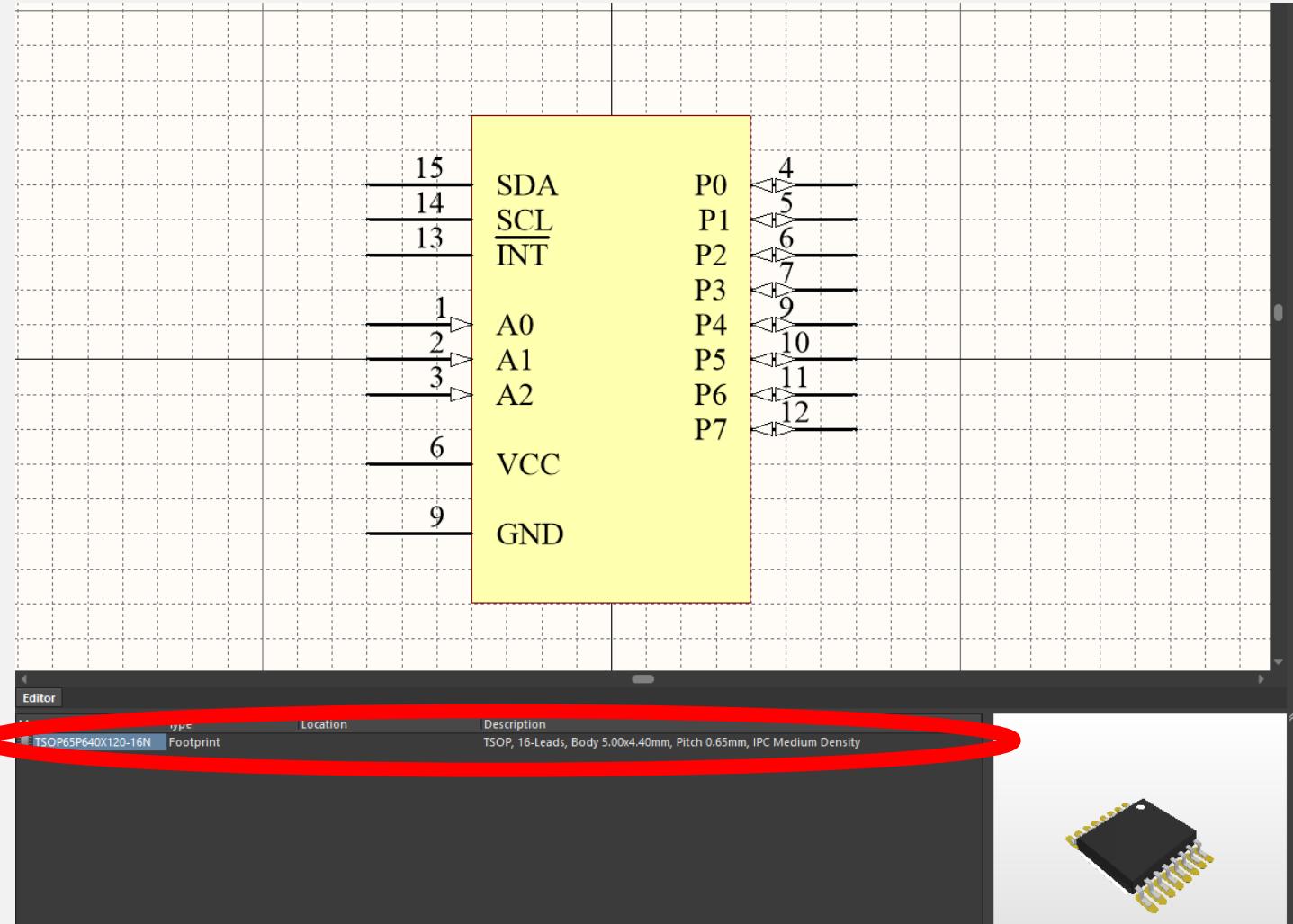


LINKING THE PCB MODEL AND THE SCHEMATIC MODEL

- You have the PCB component selected. If you hit “Pin Map” you can see how the Pin# from the Schematic symbol will relate to the Pin# of the PCB component.
- Double check that the pins are mapped correctly! Hit OK once you check.



CONGRATULATIONS



- Congratulations! You just have finished making a complete part!
- It will get quicker with practice.